INDEX

S. No	Program Name	Date	Page No	Remarks
30	Ask how many people the user wants to invite to a party. If they enter a number below 10, ask for the names and after each name display “[name] has been invited”. If they enter a number which is 10 or higher, display the message “Too many people”	01.06.22	30	
31	Set the total to 0 to start with. While the total is 50 or less, ask the user to input a number. Add that number to the total and print the message “The total is… [total]”. Stop the loop when the total is over 50.	01.06.22	31	
32	Ask the user to enter a number. Keep asking until they enter a value over 5 and then display the message “The last number you entered was a [number]” and stop the program.	01.06.22	32	
33	Ask the user to enter a number and then enter another number. Add these two numbers together and then ask if they want to add another number. If they enter “y", ask them to enter another number and keep adding numbers until they do not answer “y”. Once the loop has stopped, display the total	01.06.22	33	
34	Create a variable called compnum and set the value to 50. Ask the user to enter a number. While their guess is not the same as the compnum value, tell them if their guess is too low or too high and ask them to have another guess. If they enter the same value as compnum, display the message “Well done, you took [count] attempts”.	01.06.22	34	
35	Ask the user to enter a number between 10 and 20. If they enter a value under 10, display the message “Too low” and ask them to try again. If they enter a value above 20, display the message “Too high” and ask them to try again. Keep repeating this until they enter a value that is between 10 and 20 and then display the message “Thank you”	01.06.22	35	
36	Using the song “10 green bottles”, display the lines “There are [num] green bottles hanging on the wall, [num] green bottles hanging on the wall, and if 1 green bottle should accidentally fall”. Then ask the question “how many green bottles will be hanging on the wall?” If the user answers correctly, display the message “There will be [num] green bottles hanging on the wall”. If they answer incorrectly, display the message “No, try again” until they get it right. When the number of green bottles gets down to 0, display the message “There are no more green bottles hanging on the wall”.	01.06.22	36	
37	WAP to grade a student with percentage of 3 subject marks : a) >=75% - A b) >=60% or <75% - B c) >=40% or <60% - C d) <40% - F	01.06.22	39	

107	Write a menu driven program that reads details of a faculty. Provide options to add a new record, delete a record, update any existing record, and display all are a particular record.	17.08.22	134	
108	Program to handle the divide by zero exception	17.08.22	135	
109	Program with multiple except blocks.	17.08.22	136	
110	Program having an exception clause handling multiple exceptions simultaneously	17.08.22	137	
111	Program to demonstrate the use of except: block	17.08.22	138	
	Write a program that opens a file and writes data to it . Handle exceptions that can be generated during the I/O operations			

<u>Question 1:-</u>

Write a Python program to find the gross salary of the employee. (readBasic, Da,Hra and pf is 10% of the basic)

Aim: To find the gross salary of an employee

Description:

We input the values of salary,hra,da then we calculate the total salary and print it.

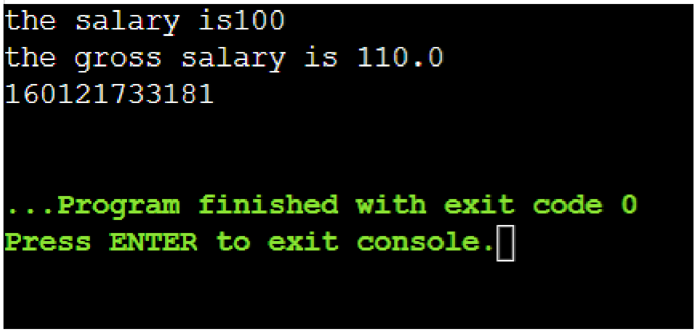
Algorithm:

```
Step-1:Start
Step-2:input the value of basic salary,da,hra
Step-3:compute pf=0.1*s
Step-4:calculate total salary and print it
Step-5:Stop
```

<u>Code</u>:

```
salary=int(input("the salary is"))
da=0.1*salary
hra=da
pf=da
gross=salary+da+hra-pf
print("the gross salary is",gross)
print(‘160121733181’)
```

Output:



```
the salary is100
the gross salary is 110.0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 2:-

Write a Python program to convert temperature from Fahrenheit to Celsius and vice versa (c=(fh-32)/1.8 and f=(c*1.8)+32

Aim: To convert temperature from Fahrenheit to Celsius and vice versa

Description:

This **Code** helps to convert degree celsius temp to farenheits and vice versa.If we entered 1 ,then we should enter celsius then we get value in farenheits if we entered 2 then vice versa happens.

Algorithm:

Step-1:Start

Step-2:Enter 1 or 2

Step-3:If n is 1

Step-4:input the value of celsius and convert it into fahrenheits and print it

Step-5:elif n is 2

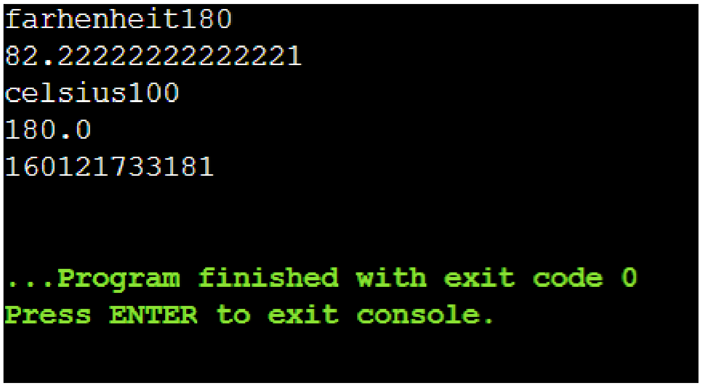
Step-6:input the value of fahrenheits and convert it into celsius and print it

Step-7:Stop

Code:

```
f=float((input("farhenheit")))

c=(f-32)/1.8

print(c)

f=float(input("celsius"))

f=(c*1.8)+32

print(f)

print('160121733181')
```

Output:



```
farhenheit180
82.22222222222221
celsius100
180.0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 3:-

Write a Python program to find area and perimeter of a circle. (area=∏r2perimeter=2∏r)

Aim: To find area and perimeter of a circle

Description:

In this **Code** we input the value of radius of a circle and find out the values of area of circle that is pi*r*r and perimeter 2*pi*r and print them.

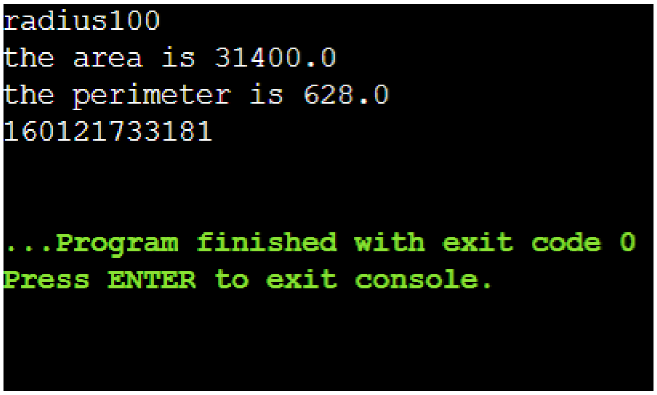
Algorithm:

```
Step-1:start
Step-2:inputting the value of radius
Step-3:computing the area value
        a=3.14*pow(r,2)
Step-4:similarly find the value of perimeter of circle
Step-5:print the area and perimeter values
Steo-6:stop
```

Code:

```
pi=3.14
r=float(input("radius"))
a=pi*r*r
print("the area is",a)
p=2*pi*r
print("the perimeter is",p)
print('160121733181')
```

Output:



```
radius100
the area is 31400.0
the perimeter is 628.0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 4:-

Write a Python Program to find least number must be added to x to obtain anumber exactly divisible by y.

Aim: To find least number must be added to x to obtain a number exactly divisible by y.

Description:

This program tells that how much we should add to x so that it is divisible by y.By inputting values of x and y from user compute the req value using formula (y-x)%y then print it.

Algorithm:

Step-1:Start

Step-2;Inputting the values of x and y

Step-3:If x%y is 0 then printing x is divisible by y

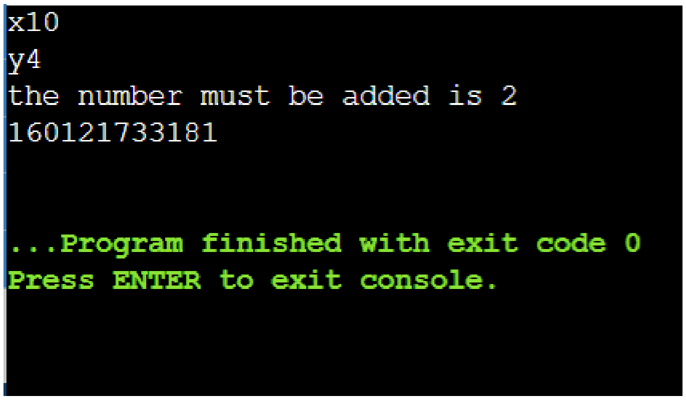
Step-4:else compute (y-x)%y and print it

Step-5:Stop

Code:

```
x=int(input("x"))

y=int(input("y"))

rem=x%y

ans=y-rem

print("the number must be added is",ans)

print('160121733181')
```

Output:



```
x10
y4
the number must be added is 2
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 5:-

Write a Python Program to find least number must be subtracted from x toget the number exactly divisible by y.

Aim: To find least number must be subtracted from x to obtain a number exactly divisible by y.

Description:

This program tells that how much we should subtract to x so that it is divisible by y.By inputting values of x and y from user compute the require value using formula (y-x)%y then print it.

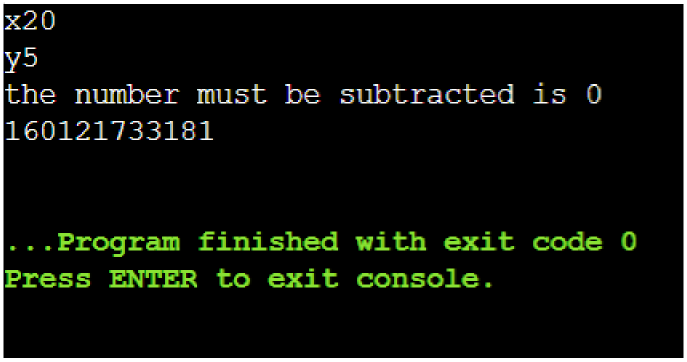
Algorithm:

```
Step-1:Start
Step-2;Inputting the values of x and y
Step-3:If x%y is 0 then printing x is divisible by y
Step-4:else compute (y+x)%y and print it
Step-5:Stop
```

Code:

```
x=int(input("x"))
y=int(input("y"))
rem=x%y
print("the number must be subtracted is",rem)
print('160121733181')
```

Output:



```
x20
y5
the number must be subtracted is 0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 6:-

Write a Python program to calculate area and perimeter of a right angledtriangle.(Base and Height are inputs)

Aim: To calculate area and perimeter of a right angled triangle

Description: In this program we have to input the values of height and base of a triangle then calculate the hypotenuse length using pythagoras theorem then calculate area and perimeter of triangle and print them

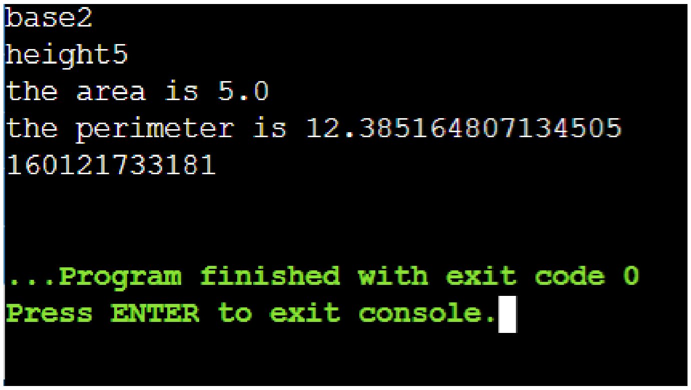
Algorithm:

```
Step-1:Start
Step-2:Input the values of base and height
Step-3:compute area from formula 0.5*b*h
Step-4:compute hypotenuse using pythogoras theorem
Step-5:find perimeter (sum of all sides) and print it
Step-6:Stop
```

Code:

```
import math
b=int(input("base"))
h=int(input("height"))
area=0.5*b*h
z=math.sqrt(b*b+h*h)
perimeter=b+h+z
print("the area is",area)
print("the perimeter is",perimeter)
print('160121733181')
```

Output:



```
base2
height5
the area is 5.0
the perimeter is 12.385164807134505
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 7:-

On dividing x by a certain number, the quotient is y and the remainder is z.Write a Python program to find the number.

Aim: To find a number such that when it divides x, the quotient is y and remainder is y

Description: Input he values of x,y,z then calculate the dividing member using formula n=(x-z)/y then print it.

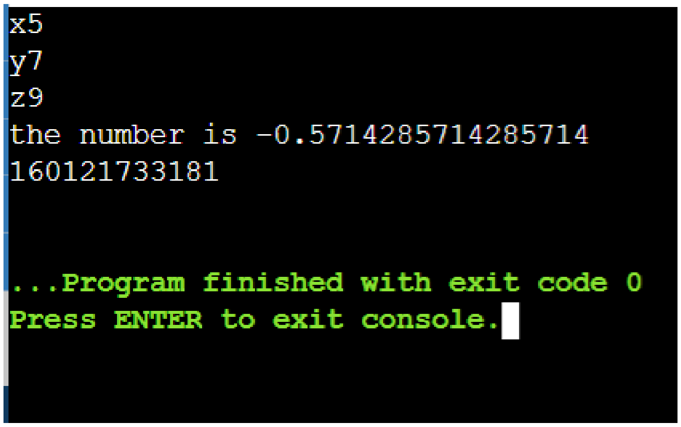
Algorithm:

```
Step-1:Start
Step-2:Input x,y,z
Step-3:compute the n value using formula n=(x-z)/y
Step-4:print n
Step-5:Stop
```

Code:

```
x=int(input("x"))
y=int(input("y"))
z=int(input("z"))
n=(x-z)/y
print("the number is",n)
print('160121733181')
```

Output:



```
x5
y7
z9
the number is -0.5714285714285714
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 8:-
Find the sum of natural numbers 1 to n.(read n as input)

Aim: To find the sum of n natural numbers

Description: Input the value of n and compute the value of n*(n+1)/2 then print it.

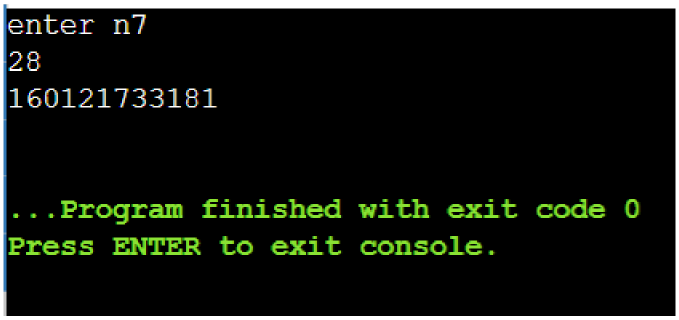
Algorithm:

```
Step-1:Start
Step-2:Inputting the n value
Step-3:Computing the value of n*(n+1)/2 then print it.
Step-4:Stop
```

Code:

```
n=int(input("enter n"))
sum=int(0)
for i in range(1,n+1):
    sum+=i
print(sum)
print('160121733181')
```

Output:



```
enter n7
28
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 9:-

Write a Python program to calculate Simple interest (SI=PTR/100)

Aim: To calculate simple interest

Description:Inputting the values of time,rate,principle amount and caluculating using p*t*r/100 then printing it.

Algorithm:

Step-1:Start

Step-2:Input the values of principle amount ,time,rate

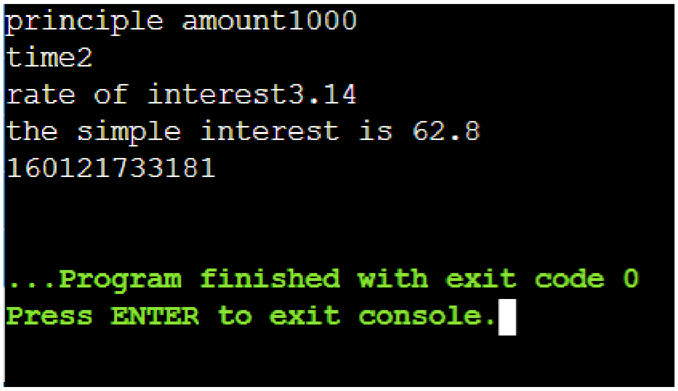
Step-3:Compute simple intrest using formula p*t*r/100 and print SI

Step-4:Stop

Code:

```
p=int(input("principle amount"))

t=int(input("time"))

r=float(input("rate of interest"))

si=p*t*r/100

print("the simple interest is",si)

print('160121733181')
```

Output:



```
principle amount1000
time2
rate of interest3.14
the simple interest is 62.8
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 10:-

Write a Python program to calculate area and perimeter of a rectangle

Aim: To calculate area and perimeter of rectangle

Description: Inputting the length and breadth values the we need to calculate the area and perimeter of the rectangle using the formula Area=Length*Breadth and Perimeter=2*(Length+Breadth).

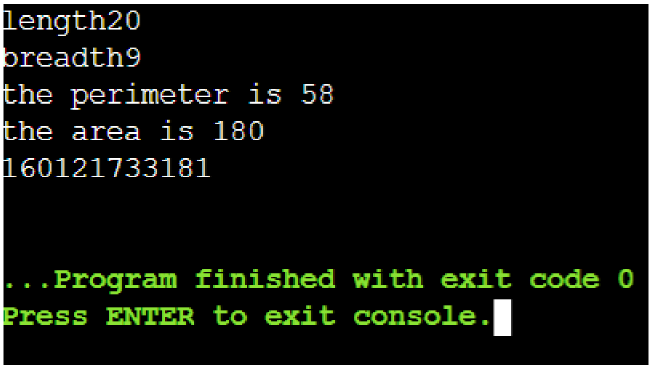
Algorithm:

```
Step-1:Start
Step-2: Input the length and breadth of the rectangle (i.e. l, b).
Step-3: Area=l*b and Perimeter=2*(l+b)
Step-4: Print the area and perimeter of the rectangle.
Step-5:stop
```

Code:

```
l=int(input("length"))

b=int(input("breadth"))

p=2*(l+b)

a=l*b

print("the perimeter is",p)

print("the area is",a)

print('160121733181')
```

Output:



```
length20
breadth9
the perimeter is 58
the area is 180
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 11:-

Write a Python program to convert centimeters to meters and meters to kilometers. (1 meter=100 centimeters 1 kilometer=1000 meters)

Aim: To convert centimetres to metres and metres to kilometres.

Description:Here we convert centimetres to meters and meters to kilometres by inputting the value of centimeters then printing meters and kilometers

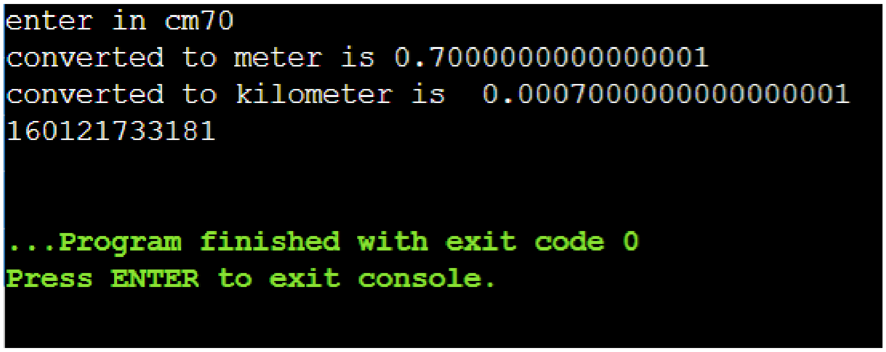
Algorithm:

```
Step-1:Start
Step-2:Input the value of centimeters
Step-3:compute  number of meters and kilometers using formula
        m=cm/100,km=m/100000
Step-4:print m,km
Step-5:Stop
```

Code:

```
x=int(input("enter in cm"))
y=0.01*x
print("converted to meter is",y)
z=0.001*y
print("converted to kilometer is ",z)
print('160121733181')
```

Output:



```
enter in cm70
converted to meter is 0.7000000000000001
converted to kilometer is  0.0007000000000000001
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 12:-

Write a Python program to convert the given numbers of days into years,

months and days. (Ignoring leap year)

Aim: To convert the given numbers of days into years, months, and days

Description:

In this **Code** we compute number of years,months and left over days can be produced from the given number of days

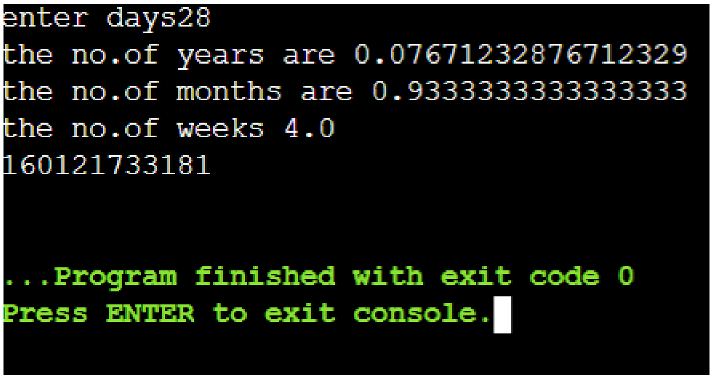
Algorithm:

```
Step-1:Start
Step-2:Input the value of days
Step-3:compute  years  from formula days//365
Step-4:compute months (days%365)//30
Step-5:compute days from (days%365)%30
Step-6:stop
```

Code:

```
x=int(input("enter days"))
y=x/365
m=x/30
w=x/7
print("the no.of years are",y)
print("the no.of months are",m)
print("the no.of weeks",w)
print('160121733181')
```

Output:



```
enter days28
the no.of years are 0.07671232876712329
the no.of months are 0.9333333333333333
the no.of weeks 4.0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 13:-

Write a Python program to calculate the angle of a triangle if two angles are

given as input

Aim: To calculate the angle of a triangle if two angles are given

Description: Here we are inputting the two angles and triangle and computing third angle by subtrating the sum of two gives angles from 180.

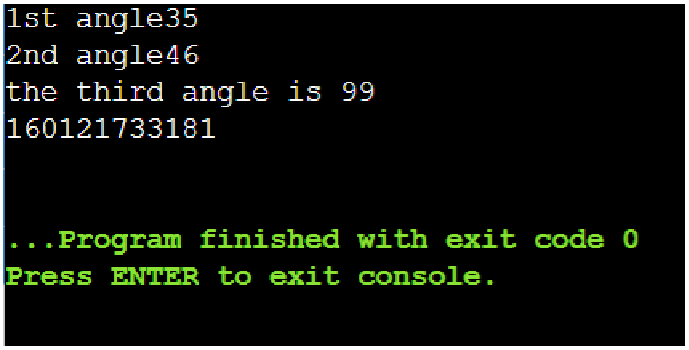
Algorithm:

```
Step-1:start
Step-2:Inputting angles a,b
Step-3:Compute  third angle c from formula c=180-a-b
Step-4:print c
Step-5:Stop
```

Code:

```
a1=int(input("1st angle"))

a2=int(input("2nd angle"))

a3=180-a1-a2

print("the third angle is",a3)

print('160121733181')
```

Output:



```
1st angle35
2nd angle46
the third angle is 99
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 14:-

Write a Python program to read the consumer number and number of units consumed and the cost per unit and print the amount to be paid. (Amt=num of units*cost)

Aim: To read the consumer number and number of units consumed and the cost per unit and print the amount to be paid

Description: In this program we input the values of number of units and cost per unit then calculate the Amt and printing it.

Algorithm:

Step-1:Start

Step-2:inputing the value of number of units consumed and the cost per unit

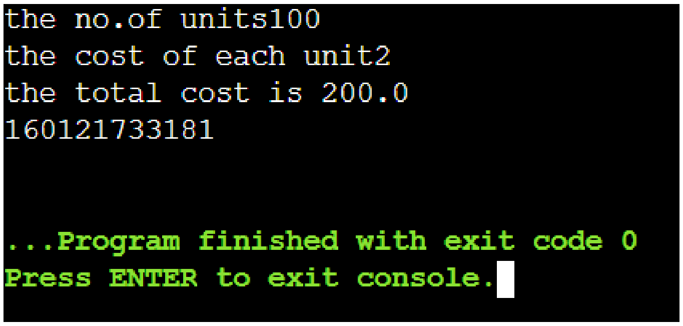
Step-3:Calculating amt and printing it

Step-4;Stop

Code:

```
n=int(input('the no.of units'))

c=float(input('the cost of each unit'))

tc=n*c

print('the total cost is',tc)

print('160121733181')
```

Output:



```
the no.of units100
the cost of each unit2
the total cost is 200.0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 15:-

Write a Python program to find addition of two numbers using subtraction operator. Sum=a-(-b)

Aim: To find addition of two numbers using subtraction operator

Description:

Input two numbers to add them using '-' we can do like this a-(-b)

Algorithm:

Step-1:Start

Step-2:Input two numbers

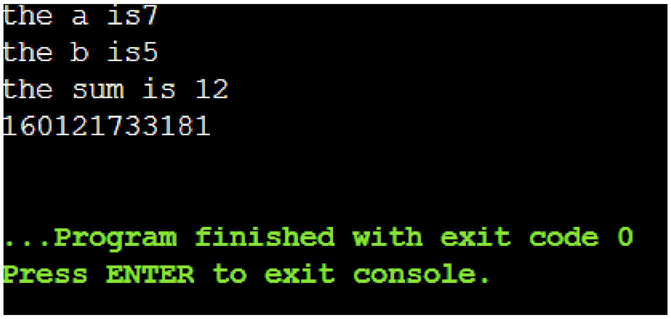
Step-3:compute a-(-b) and print it.

Step-4:Stop

Code:

```
a=int(input('the a is'))
b=int(input('the b is'))
s=a-(-b)
print('the sum is',s)
print('160121733181')
```

Output:



```
the a is7
the b is5
the sum is 12
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 16:-

Write a Python Program to calculate area and perimeter of a triangle giveninput as three sides a,b,c. Perimeter=(a+b+c) s=(a+b+c)/2 Area=sqrt(s*(s- a)(s-b)*(s-c))

Aim: to calculate area and perimeter of a triangle given input as three sides a,b,c

Description: Inputting the sides of a triangle a,b,c computing the perimeter from formula a+b+c and area from sqrt(s*(s-a)*(s-b)*(s-c)) where s is half of perimeter

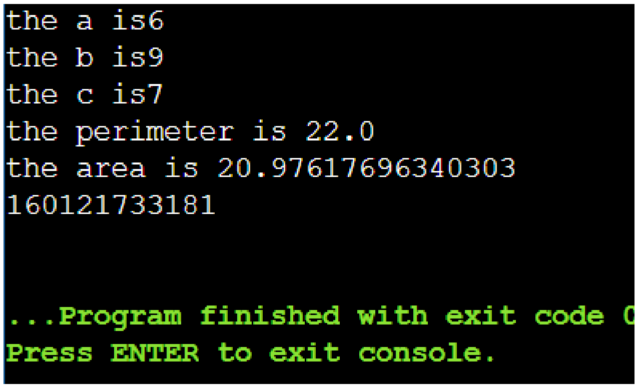
Algorithm:

```
Step-1:Start
Step-2:Input the sides of triangle a,b,c
Step-3:compute s=(a+b+c)/2
Step-4:print perimeter
Step-5:compute area from formula sqrt(s*(s-a)*(s-b)*(s-c)) and print it
Step-6:Stop
```

Code:

```
import math
a=int(input('the a is'))
b=int(input('the b is'))
c=int(input('the c is'))
s=(a+b+c)/2
a=math.sqrt(s*(s-a)*(s-b)*(s-c))
p=2*s
print('the perimeter is',p)
print('the area is',a)
print('160121733181')
```

Output:



```
the a is6
the b is9
the c is7
the perimeter is 22.0
the area is 20.97617696340303
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 17:-

Write a Python program to read five Subject marks and find the average.

Aim: To read five Subject marks and find the average.

Description: Input the five subjects marks and find avg of the and print it

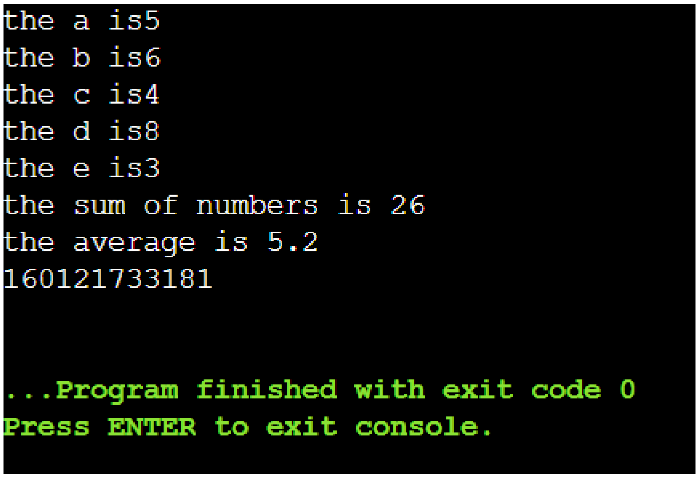
Algorithm:

```
Step-1:Start
Step-2:Input the marks of five subjects
Step-3:find average of five subjects and print the avg
Step-4:Stop
```

Code:

```
import math
a=int(input('the a is'))
b=int(input('the b is'))
c=int(input('the c is'))
d=int(input('the d is'))
e=int(input('the e is'))
s=(a+b+c+d+e)
print('the sum of numbers is',s)
a=s/5
print('the average is',a)
print('160121733181')
```

Output:



```
the a is5
the b is6
the c is4
the d is8
the e is3
the sum of numbers is 26
the average is 5.2
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 18:-

Write a Python Write a program to read the values of x, y and z and print the results of the following expressions in one line. 1. (x+y+z) / (x-y-z) 2. (x+y+z) /3 3. (x+y) * (x-y) * (y-z)

Aim: To read the values of x, y and z and print the results of the expressions in one line

Description: We need to calculate the required values using the formulas given in the question

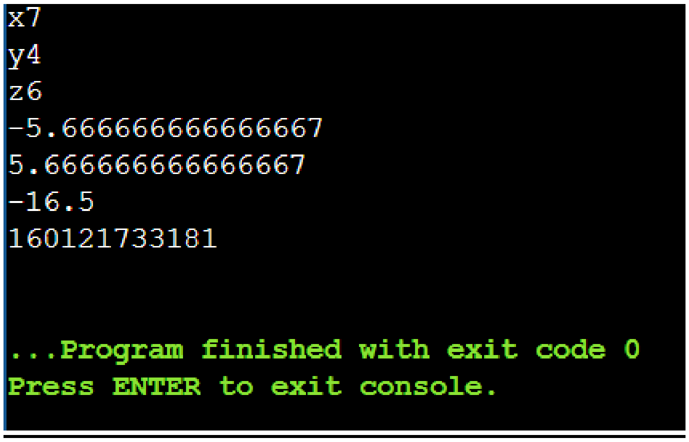
Algorithm:

```
Step-1:Start
Step-2: Input x, y, z
Step-3: a=(x+y+z)/(x-y-z) , b=(x+y+z)/3, c=(x+y)*(x-y)*(y-z)
Step-4: Print a, b, c
Step-5:Stop
```

Code:

```
x=int(input('x'))

y=int(input('y'))

z=int(input('z'))

a=(x+y+z)/(x-y-z)

print(a)

b=(x+y+z)/3

print(b)

c=(x+y)*(x-y)/(y-z)

print (c)

print('160121733181')
```

Output:



```
x7
y4
z6
-5.666666666666667
5.666666666666667
-16.5
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 19:-

Write a Python program to Calculate Compound interest CI=p(1+r/100)n(Inputs are p=principle amount ,r=interest rate, n=no. of years)

Aim: To Calculate Compound interest

Description: Calculating CI=p(1+r/100)^n by inputting values of rate,amt,no.of.years

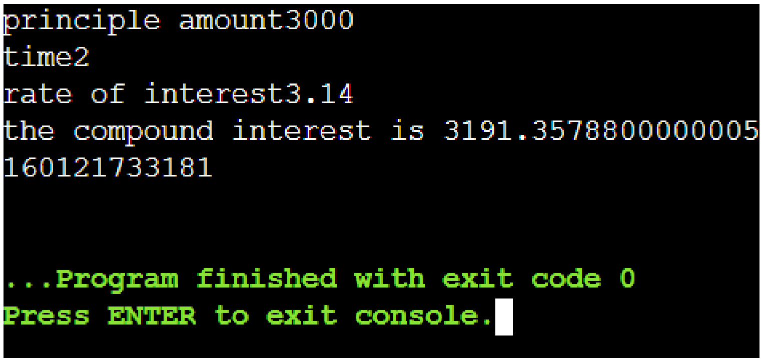
Algorithm:

```
Step-1:start
Step-2:Input the values of rate,principle amount, no.of.years
Step-3:compute CI=p(1+r/100)^n and print CI
Step-4:stop
```

Code:

```
p=int(input('principle amount'))
n=int(input('time'))
r=float(input('rate of interest'))
ci=p*pow(1+r/100,n)
print('the compound interest is',ci)
print('160121733181')
```

Output:



```
principle amount3000
time2
rate of interest3.14
the compound interest is 3191.3578800000005
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 20:-

The displacement of a body (s) against time (t) where (u) is the initial velocity and (a) is the acceleration is given by the formula S = ut + ½ at2 . Write a Python Code to find S for given u , t , a.

Aim: to find displacement for given initial velocity, time , acceleration

Description: We need to calculate the displacement covered by a body from the acceleration, time and initial speed by the formula S = ut + ½ at^2

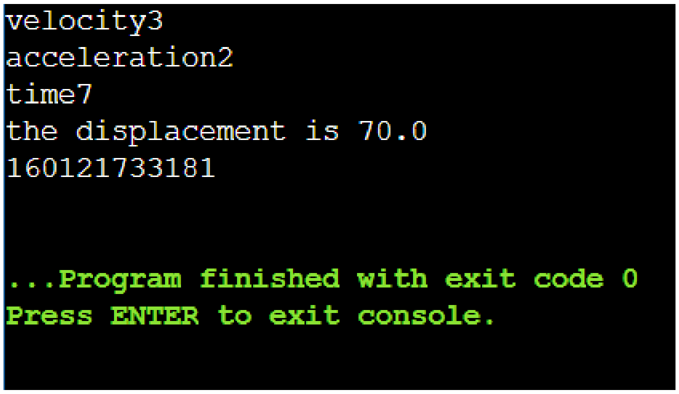
Algorithm:

```
Step-1: Input the acceleration, initial speed, and time
Step-2: S = ut + ½ at^2
Step-3: Print S
```

Code:

```
u=int(input('velocity'))

a=int(input('acceleration'))

t=int(input('time'))

s=u*t+0.5*a*(t**2)

print('the displacement is',s)

print('160121733181')
```

Output:



```
velocity3
acceleration2
time7
the displacement is 70.0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 21:-

Ask the user to enter their name and then display their name three time

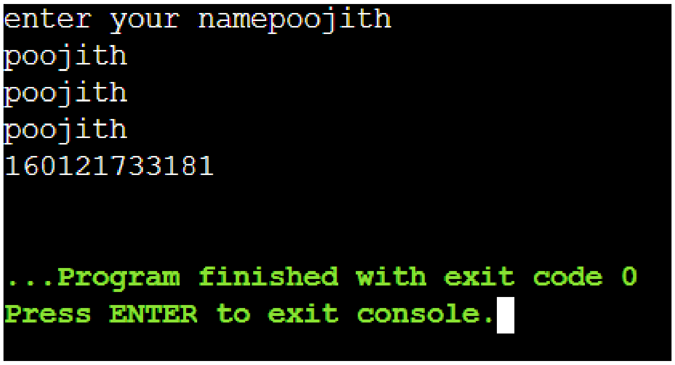
Aim: to enter user name and then display their name three times

Description: Inputting user name and printing it 3 times using loop

Code:

```
name=input("enter your name")
for i in range(3):

    print(name)

print('160121733181')
```

Output:



```
enter your namepoojith
poojith
poojith
poojith
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 22:-

Alter the above program so that it will ask the user to enter their name and anumber and then display their name that number of times.

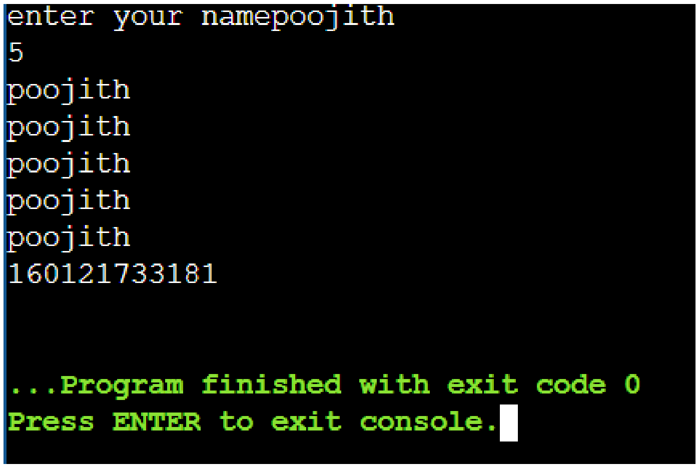
Aim: To enter user name and a number and then display their name that number of times.

Description: Altering the above program that it also asks the number of iterations to user

Code:

```
name=input("enter your name")

n=int(input())

for i in range(n):

    print(name)

print('160121733181')
```

Output:



```
enter your namepoojith
5
poojith
poojith
poojith
poojith
poojith
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 23:-

Ask the user to enter their name and display each letter in their name on a separate line

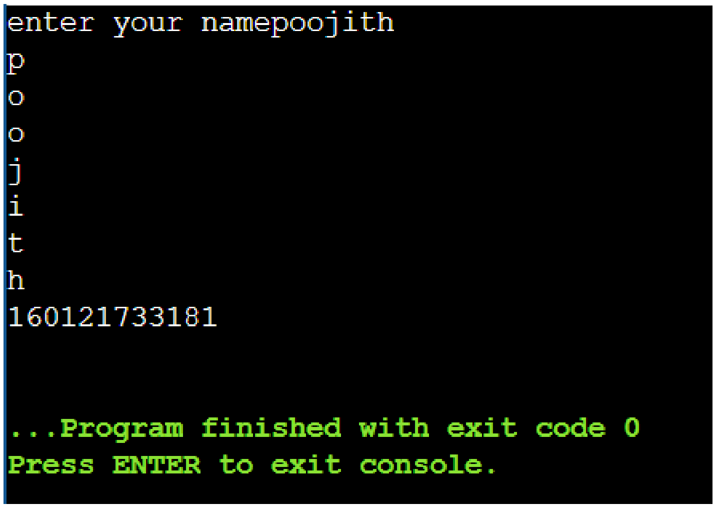
Aim: To enter their name and display each letter in their name on a separate line.

Description:Inputting user name and printing each letter of user in each line using looping statements so that each letter will be in single line

Code:

```
name=input("enter your name")
for i in name:
   print(i)
print('160121733181')
```

Output:



```
enter your namepoojith
p
o
o
j
i
t
h
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 24:-

Change above program to also ask for a number. Display their name (oneletter at a time on each line) and repeat this for the number of times they entered.

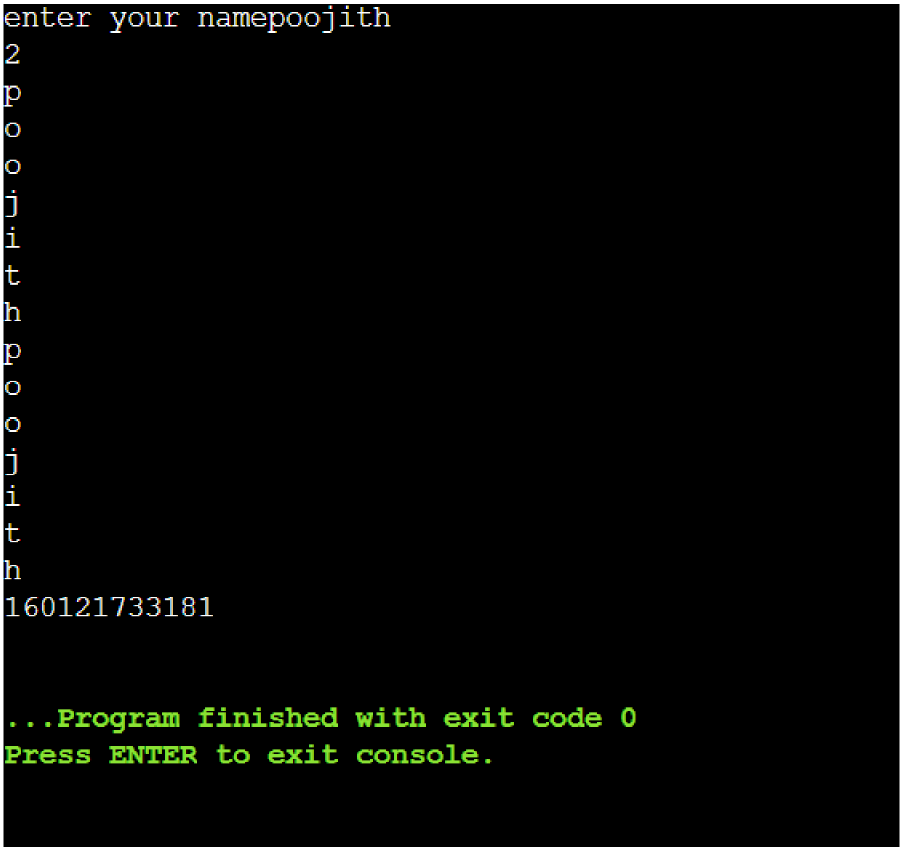
Aim: To enter user name and number and display each letter in name for number of times entered

Description: Inputting user name and asking number of iterations .printing his letter in each line of number of times user asked

Code:

```
name=input("enter your name")

n=int(input())

for i in range(n):

    for j in name:

        print(j)

print('160121733181')
```

Output:



```
enter your namepoojith
2
p
o
o
j
i
t
h
p
o
o
j
i
t
h
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 25:-

Ask the user to enter a number between 1 and 12 and then display the timestable for that number.

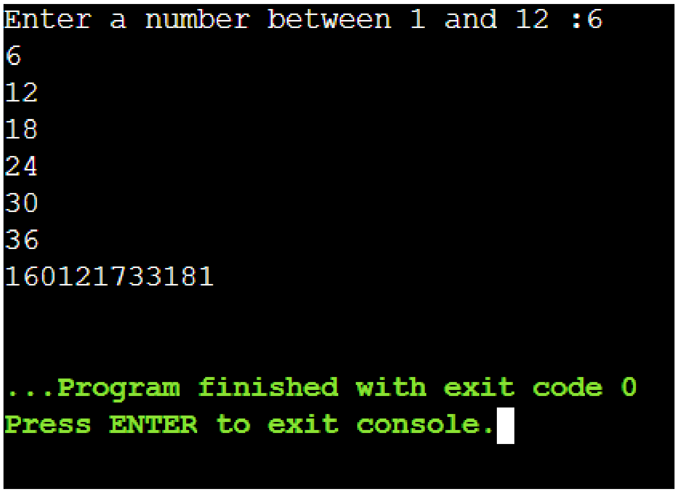
Aim: To enter a number between 1 and 12 and then display the times table for that number

Description:Inputting a number in between 1 and 12 and printing the times table of it

Code:

```
num = int(input('Enter a number between 1 and 12 :'))
for i in range(1,num+1) :
    print(num*i)
print('160121733181')
```

Output:



```
Enter a number between 1 and 12 :6
6
12
18
24
30
36
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 26:-

Ask for a number below 50 and then count down from 50 to that number, making sure you show the number they entered in the Output.

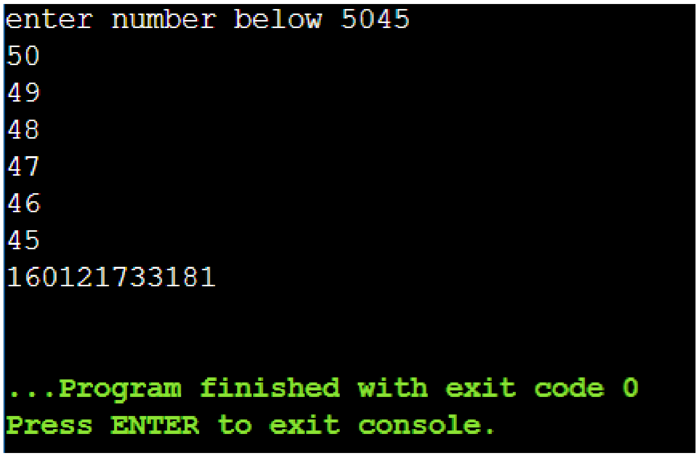
Aim: To enter a number below 50 and count down from 50 to that number

Description:Inputting a number below 50 then printing numbers from 50 to that number

Code:-

```
n=int(input("enter number below 50"))

t=50

while(t>=n):

    print(t)

    t=t-1

print('160121733181')
```

Output:



```
enter number below 5045
50
49
48
47
46
45
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 27:-

Ask the user to enter their name and a number. If the number is less than10, then display their name that number of times; otherwise display the message “Too high” three times.

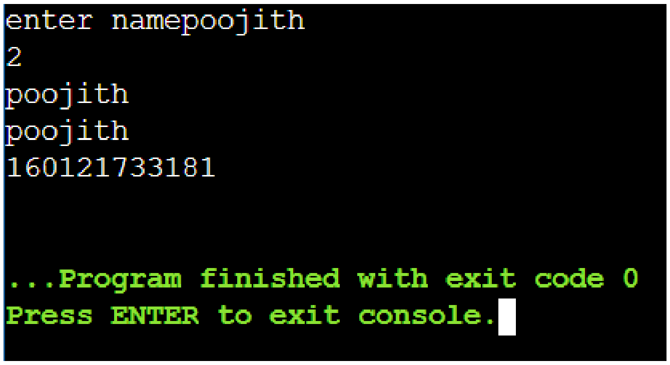
Aim: To enter user name and number and apply certain conditions on that number

Description:Input user name and a number .if the number is <10 then print the user name that many times else print too high 3 times

Code:

```
name=input("enter name")

n=int(input())

if (n<10):

    for i in range(n):

        print(name)

elif(n>10):

    for j in range(3):

        print("Too high")

print('160121733181')
```

Output:



```
enter namepoojith
2
poojith
poojith
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 28:-

Set a variable called total to 0. Ask the user to enter five numbers and after each input ask them if they want that number included. If they do, then addthe number to the total. If they do not want it included, don’t add it to the total. After they have entered all five numbers, display the total.

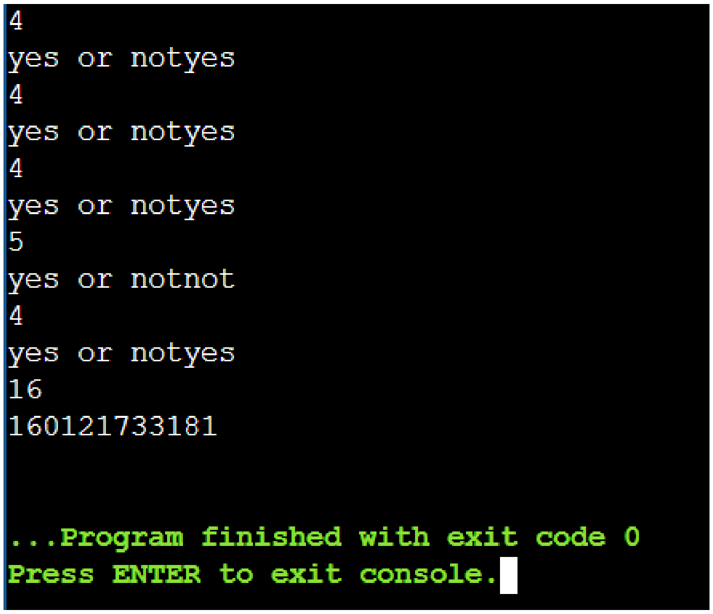
Aim: To enter 5 numbers and add the required numbers

Description: Inputting 5 numbers from user on each inputting we will ask user whether he want to include that number to include in sum or not if yes we include in total else no.Finally we print the total

Code:

```
total=0
for i in range(5):
    k=int(input())
    c=input("yes or not")
    if (c=="yes"):
        total+=k
    else:
        total=total
print(total)
print('160121733181')
```

Output:



```
4
yes or notyes
4
yes or notyes
4
yes or notyes
5
yes or notnot
4
yes or notyes
16
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 29:-

Ask which direction the user wants to count (up or down). If they select up,then ask them for the top number and then count from 1 to that number. Ifthey select down, ask them to enter a number below 20 and then count down from 20 to that number. If they entered something other than up or down, display the message “I don’t understand”.

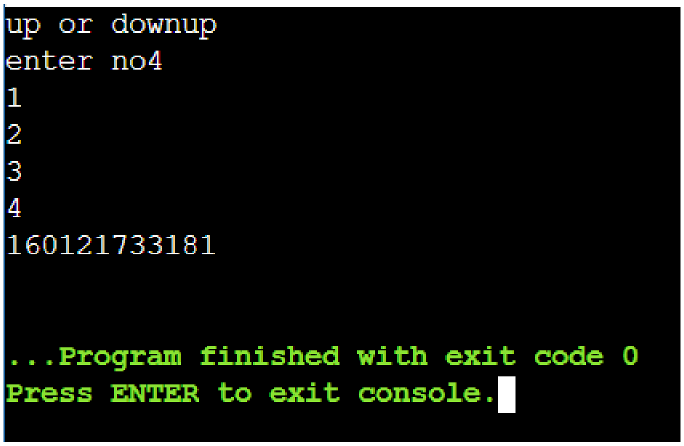
Aim: To apply certain conditions to a user input

Description: In this program we need to ask the used to enter if he wants to count upwards or downwards and need to count it from 1 if it is up till the number and from 20 if it is down and if the input is something other than yes or no print I Don’t Understand.

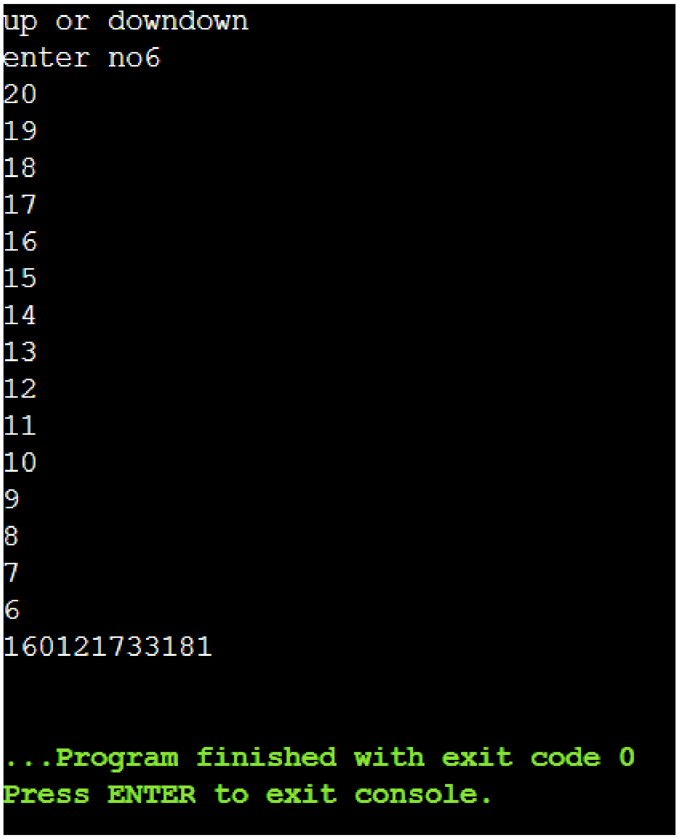
Code:

```
c=input("up or down")
if (c=="up"):
   n=int(input("enter no"))
   for i in range(1,n+1):
      print(i)
elif(c=="down"):
   m=int(input("enter no"))
   t=20
   while(t>=m):
      print(t)
      t=t-1
else:
   print("I don't understand")
print('160121733181')
```

Output:



```
up or downup
enter no4
1
2
3
4
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```



```
up or downdown
enter no6
20
19
18
17
16
15
14
13
12
11
10
9
8
7
6
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 30:-

Ask how many people the user wants to invite to a party. If they enter a number below 10, ask for the names and after each name display “[name] has been invited”. If they enter a number which is 10 or higher, display themessage “Too many people”

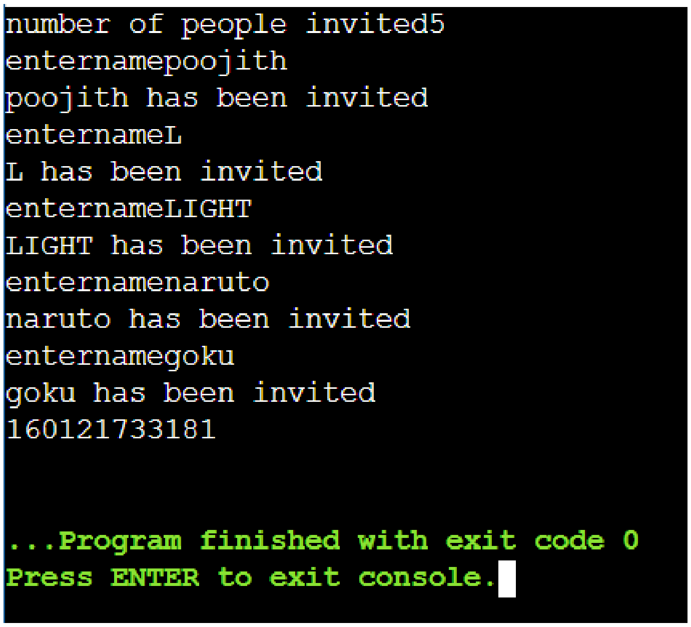
Aim: To enter number and name of people invited for party and **Output** name of the person

Description: For a party some guests are invited .If number of guests is <10 then asks their names and print that the guest has been invited else print too man people.

Code:

```
n=int(input("number of people invited"))
if (n<10):
   for i in range(n):
       name=input("entername")
       print(name,"has been invited")
else:
   print("Too many people invited")
print('160121733181')
```

Output:



```
number of people invited5
enternamepoojith
poojith has been invited
enternameL
L has been invited
enternameLIGHT
LIGHT has been invited
enternamenaruto
naruto has been invited
enternamegoku
goku has been invited
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 31:-

Set the total to 0 to start with. While the total is 50 or less, ask the user toinput a number. Add that number to the total and print the message “Thetotal is… [total]”. Stop the loop when the total is over 50.

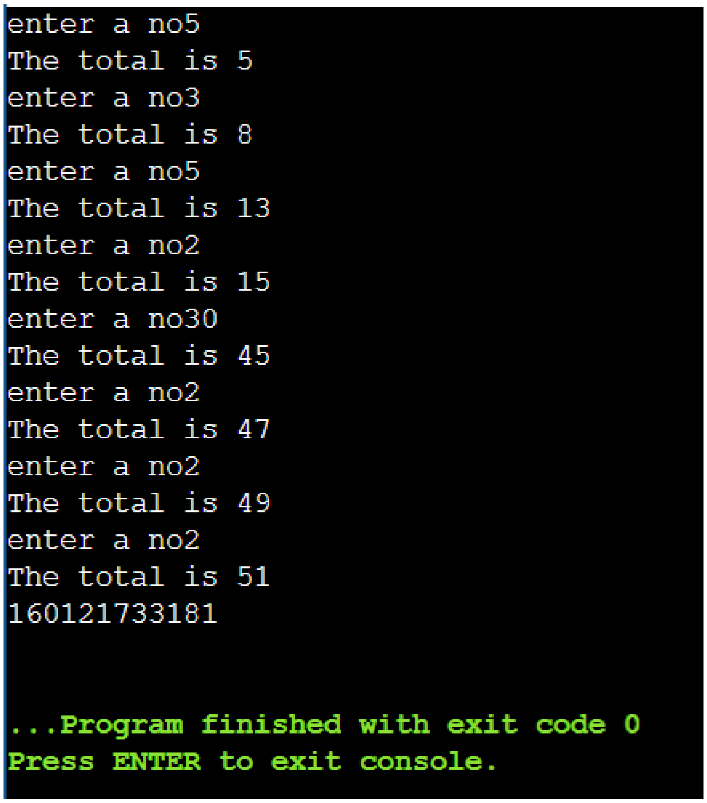
Aim: To keep entering numbers and adding them until sum crosses 50

Description: Setting total 0 and inputting numbers continuosly till the total is 50 and in after each addition to total print the total

Code:

```
total=0
while(1):
   n=int(input("enter a no"))
   total+=n
   print("The total is",total)
   if(total>50):
      break
print('160121733181')
```

Output:



```
enter a no5
The total is 5
enter a no3
The total is 8
enter a no5
The total is 13
enter a no2
The total is 15
enter a no30
The total is 45
enter a no2
The total is 47
enter a no2
The total is 49
enter a no2
The total is 51
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 32:-

Ask the user to enter a number. Keep asking until they enter a value over 5and then display the message "The last number you entered was a [number]" and stop the program.

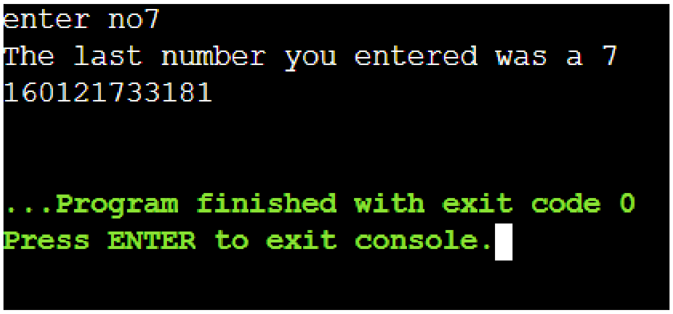
Aim: To enter a number until a number over 5 is encountered.

Description: Running and infinite loop by taking inputs continuosly until user gives a value>5 and also after every entering of input display the last entered number.

Code:-

```
while(1):
   n=int(input("enter no"))
   print("The last number you entered was a",n)
   if(n>5):
      break
print('160121733181')
```

Output:



```
enter no7
The last number you entered was a 7
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 33:-

Ask the user to enter a number and then enter another number. Add these two numbers together and then ask if they want to add another number. If they enter “y", ask them to enter another number and keep adding numbersuntil they do not answer “y”. Once the loop has stopped, display the total

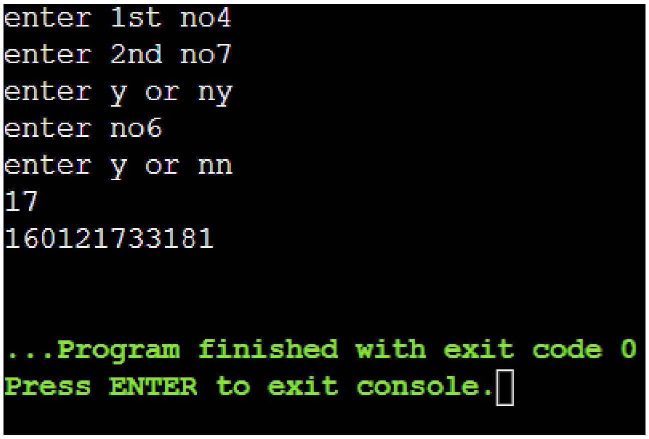
Aim: To keep adding user defined numbers until user stops inputting numbers

Description: In this program we input two numbers and added together then continuosly inuptting values by asking user and adding to the total if the user enters other than y then we exit from loop and printing the total

Code:-

```
m=int(input("enter 1st no"))

n=int(input("enter 2nd no"))

add=m+n

while(1):

   c=input("enter y or n")

   if (c=="y"):

       k=int(input("enter no"))

       add+=k

   else:

       break

print(add)

print('160121733181')
```

Output:



```
enter 1st no4
enter 2nd no7
enter y or ny
enter no6
enter y or nn
17
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 34:-

Create a variable called compnum and set the value to 50. Ask the user to enter a number. While their guess is not the same as the compnum value, tell them if their guess is too low or too high and ask them to have another guess. If they enter the same value as compnum, display the message “Welldone, you took [count] attempts”.

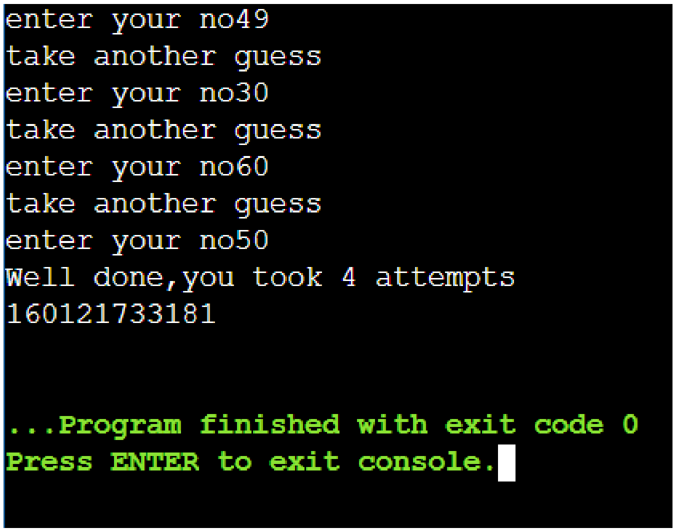
Aim: To set a value to a variable and enable user to keep guessing until guess value is same as variable

Description: Fix a number as 50.Now with infinite loop take inputs if the input is less than 50 then print too low or the input is >50 print too high if it is equal print no.of attempts taken to guess and break the loop

Code:

```
compnum=50
count=1
while(1):
   n=int(input("enter your no"))
   if (n<50):
      print("take another guess")
      count+=1
   elif(n>50):
      print("take another guess")
      count+=1
   else:
      print("Well done,you took",count,"attempts")
      break
print('160121733181')
```

Output:



```
enter your no49
take another guess
enter your no30
take another guess
enter your no60
take another guess
enter your no50
Well done,you took 4 attempts
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 35:-

Ask the user to enter a number between 10 and 20. If they enter a value under 10, display the message “Too low” and ask them to try again. If they enter a value above 20, display the message “Too high” and ask them to try again. Keep repeating this until they enter a value that is between 10 and 20and then display the message “Thank you”

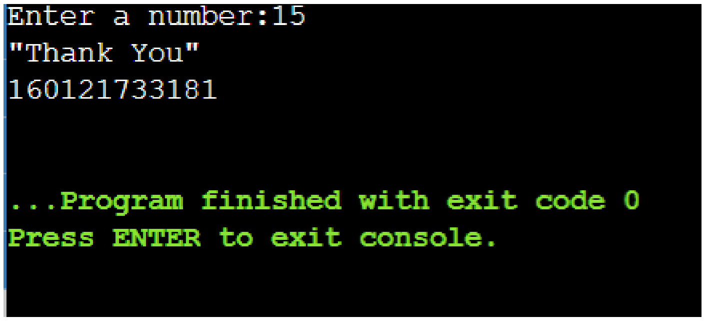
Aim: To enable user to keep entering numbers until the value entered lies between 10 and 20

Description: Inputting a number from user in infinite loop ,if the number is >20 print too high if it is <10 print too low else print thank you and break the infinite loop

Code:

```
while(1):
    n=int(input("Enter a number:"))
    if n<10:
        print("\"Too Low\"")
    elif n>20:
        print("\"Too High\"")
    else:
        print("\"Thank You\"")
        break
print('160121733181')
```

Output:



```
Enter a number:15
"Thank You"
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

<u>Question 36:-</u>

Using the song "10 green bottles", display the lines "There are [num] green bottles hanging on the wall, [num] green bottles hanging on the wall, and if 1green bottle should accidentally fall". Then ask the question "how many green bottles will be hanging on the wall?" If the user answers correctly, display the message "There will be [num] green bottles hanging on the wall".If they answer incorrectly, display the message "No, try again" until they get it right. When the number of green bottles gets down to 0, display the message "There are no more green bottles hanging on the wall".

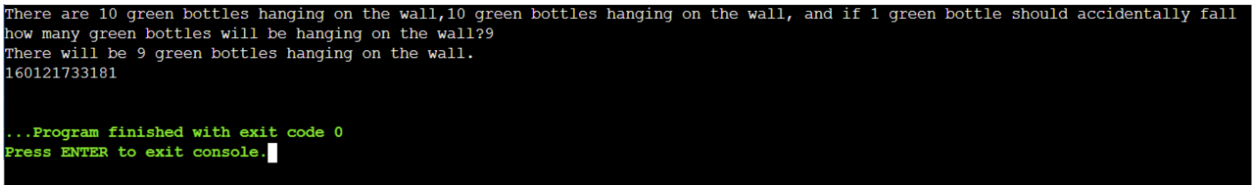
Aim: To ask user to answer question based on certain condition until the condition is satisfied

Description: In this program we need to assume 10 bottles hanging initially and ask the user how many are remaining as 1 breaks and if the user replies correctly we need to move further till no bottle remains else we need to ask the user to try again till he answers correctly.

<u>Code</u>:

```
i=10

while(1):

   print(f"There are {i} green bottles hanging on the wall,{i} green bottles hanging on the wall,
and if 1 green bottle should accidentally fall")

   n=int(input("how many green bottles will be hanging on the wall?"))

   if n==i-1:

      i=i-1

      print("There will be {0} green bottles hanging on the wall.".format(i))

      break

   elif i<1:

      print("There are no more green bottles hanging on the wall.")

      break

   else:

      print("No, Try again.")


print('160121733181')
```

Output:



```
There are 10 green bottles hanging on the wall,10 green bottles hanging on the wall, and if 1 green bottle should accidentally fall
how many green bottles will be hanging on the wall?9
There will be 9 green bottles hanging on the wall.
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 37:-

WAP to grade a student with percentage of 3 subject marks : a) >=75% - A b) >=60% or <75% - B c) >=40% or <60% - C d) <40% - F

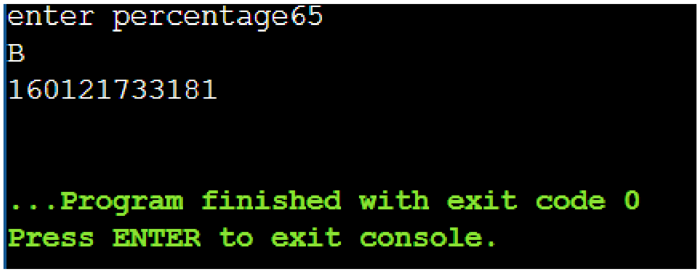
Aim: To grade a student with percentage of 3 subject marks

Description: Find percentage of a students from 3 subjects and form given conditions print the required Output.

Code:

```
p=int(input("enter percentage"))
if(p>=75):
   print("A")
elif(p>=60 and p<75):
   print("B")
elif(p>=40 and p<60):
   print("C")
else:
   print("F")
print('160121733181')
```

Output:



```
enter percentage65
B
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 38:-

WAP to identify a) Armstrong number b) palindrome for given

Aim: To identify whether a number is Armstrong number or a Palindrome number

Description: Armstrong number is a number if sumof the cubes of the digits is equal to the number thenit is armstrong number .In this program we check whether the number is armstrong or not.

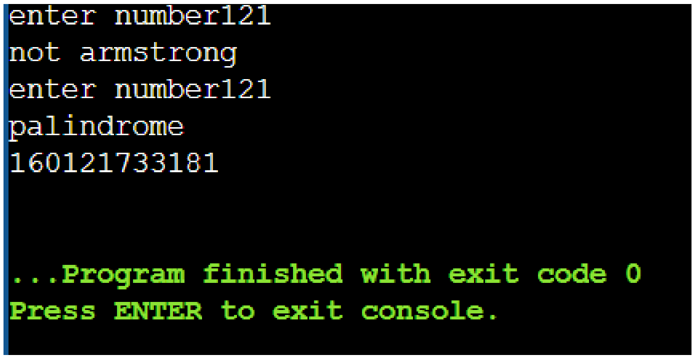
Description:A number if reversed is also same then it is palindrome

Code:

```
n=int(input("enter number"))
t=n
rem=0
sum=0
while(n>0):
   rem=n%10
   sum+=rem**3
   n=n//10
if (sum==t):
   print("armstrong")
else:
   print("not armstrong")
n=int(input("enter number"))
t=n
rem=0
sum=0
while(n>0):
   rem=n%10
```

```
    sum=sum*10+rem
    n=n//10
if (sum==t):
    print("palindrome")
else:
    print("not palindrome")
print('160121733181')
```

Output:



```
enter number121
not armstrong
enter number121
palindrome
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 39:

WAP to generate prime numbers within the given range.

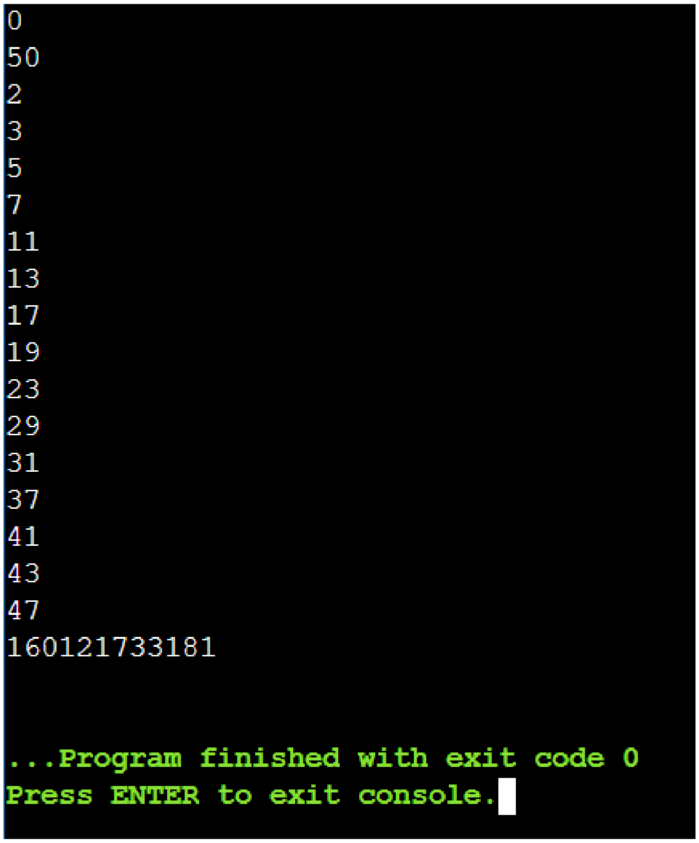
Aim: To generate prime numbers between two given numbers

Description: Printing the prime numbers between the given numbers

Code:

```
n1=int(input())
n2=int(input())
for num in range(n1,n2+1):
    if num>1:
        for i in range(2,num):
            if(num%i==0):
                break
        else:
            print(num)
print('160121733181')
```

Output:



```
0
50
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 40:

WAP to generate Fibonacci series up to given n value.

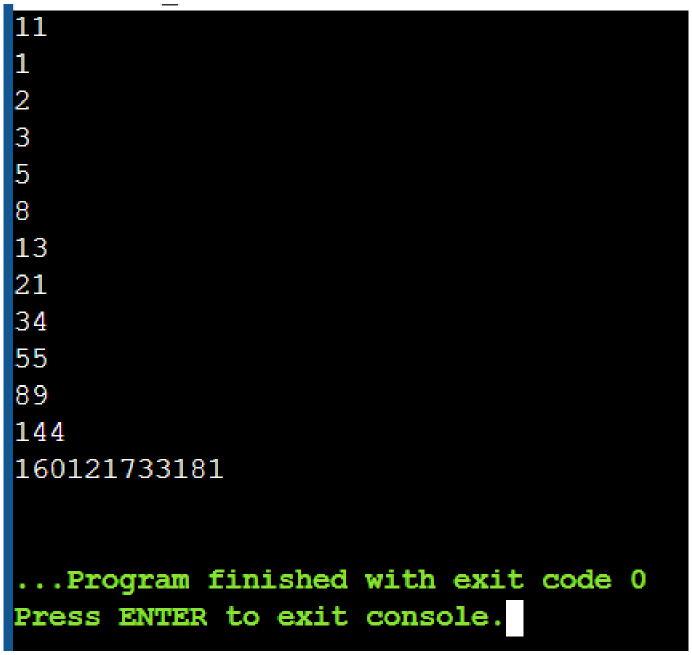
Aim: To generate Fibonacci series up to a given n value

Description: We need to print fibonacci series in this program .0 1 1 2 3 5 …..is called as fibonacci series here sum of previous two elements is equal to the present working element so by this logic we build the Code.

Code:

```
n=int(input())

i=0

j=1

for m in range(n):

   k=i+j

   i=j

   j=k

   print(k)

print('160121733181')
```

Output:



```
11
1
2
3
5
8
13
21
34
55
89
144
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 41:-

WAP to demonstrate break, continue and pass statements.

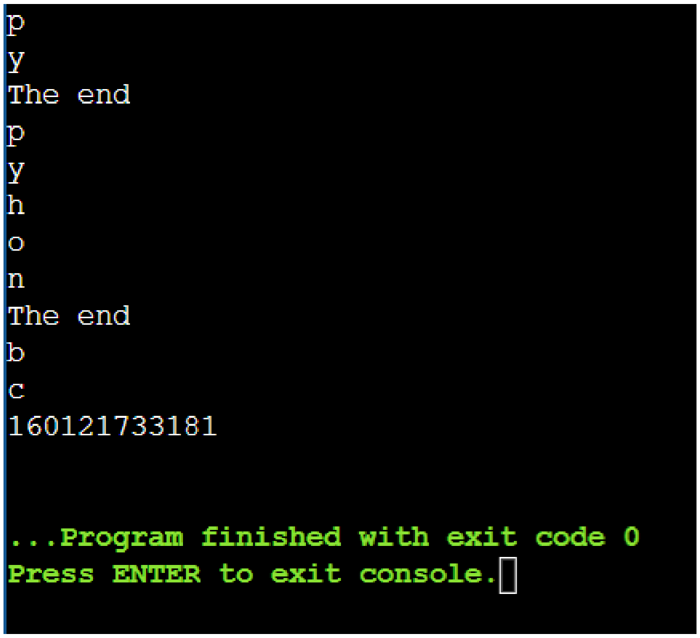
Aim: To demonstrate break, continue and pass statements

Description: In this program we just explain the use of break,continue ,pass statements

Code:

```
for val in "python":
    if val=="t":
        break
    print(val)
print("The end")
for val in "python":
    if val=="t":
        continue
    print(val)
print("The end")
li=['a','b','c']
for i in li:
    if(i=='a'):
        pass
    else:
        print(i)
print('160121733181')
```

Output:



```
p
y
The end
p
y
h
o
n
The end
b
c
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

<u>Question 42,43:-</u>

A) **Write a Python program to sum all the items in a list.**
B) **Write a Python program to get the largest number from a list**

Aim: To sum all elements of a list

Description: In this program we use sum function to find sum of all elements of the list

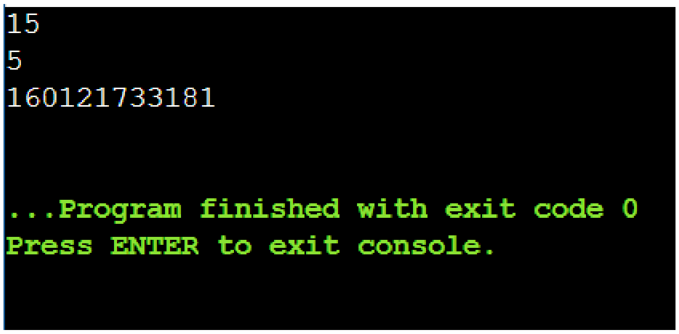
Aim: To get the largest number from a list

Description: In this program with help of max function we find out the maximum element of list and print it down

<u>Code</u>:

```
list=[1,2,3,4,5]

print(sum(list))

print(max(list))

print('160121733181')
```

Output:



```
15
5
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

<u>Question 44:-</u>

Write a Python program to check whether an element exists within a tuple.

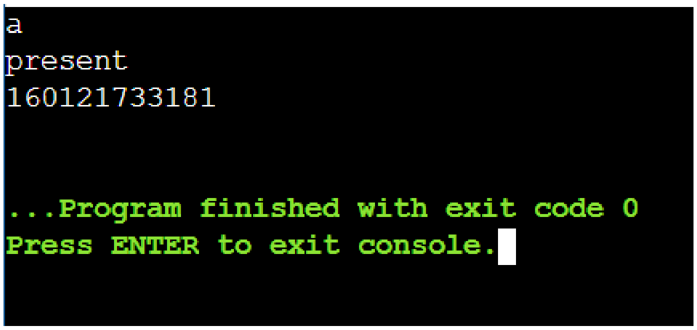
Aim: To check whether an element exists within a tuple

Description: In this program we are basically searching for a key element in tuple we here use membership operators to get Output.

<u>Code</u>:

```
t=("a","b","c")
ele=input()
for i in t:
    if i==ele:
        print("present")
print('160121733181')
```

Output:



```
a
present
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 45:-

Write a Python program to find the key of the maximum value in a dictionary. Sample Output: Original dictionary elements: {'Theodore': 19, 'Roxanne': 22, 'Mathew': 21, 'Betty': 20} Finds the key of the maximum andminimum value of the said dictionary: ('Roxanne', 'Theodore')

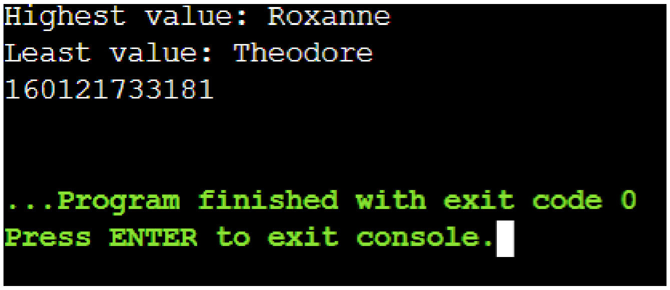
Aim: To find the maximum and minimum value of a given dictionary

Description:In this Code we need to print name of the persons who have max and min age ,with help of finding index of keys list we get max and min age persons

Code:

```
d={'Theodore': 19, 'Roxanne': 22, 'Mathew': 21, 'Betty': 20}

l=d.values()

a=max(l)

b=min(l)

for i in d:

    if d[i]==a:

        print("Highest value:",i)

for i in d:

    if d[i]==b:

        print("Least value:",i)

print('160121733181')
```

Output:



```
Highest value: Roxanne
Least value: Theodore
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 46:-

Write a Python program to extract a list of values from a given list of dictionaries. Original Dictionary: [{'Math': 90, 'Science': 92}, {'Math': 89, 'Science': 94}, {'Math': 92, 'Science': 88}] Extract a list of values from said list of dictionaries where subject = Science [92, 94, 88] Original Dictionary: [{'Math': 90, 'Science': 92}, {'Math': 89, 'Science': 94}, {'Math': 92, 'Science': 88}] Extract a list of values from said list of dictionaries where subject = Math[90, 89, 92]

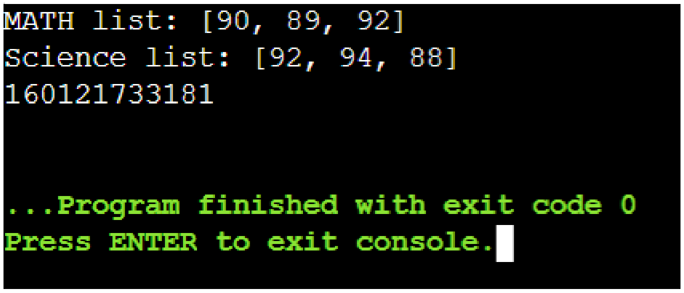
Aim: To get a list of values from list of dictionaries for a particular key

Description: In this program we need to print the marks of all the students in each subject separately in the form of a dictionary.

Code:

```
d=[{'Math': 90, 'Science': 92}, {'Math': 89, 'Science': 94}, {'Math': 92, 'Science': 88}]
l=[]
b=[]
for i in d:
    for j in i:
      if j=='Math':
          l.append(i[j])
for i in d:
    for j in i:
      if j=='Science':
          b.append(i[j])
print("MATH list:",l)
print("Science list:",b)
print('160121733181')
```

Output:



```
MATH list: [90, 89, 92]
Science list: [92, 94, 88]
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 47 :

Write a program to determine whether the character entered is a vowel ornot.

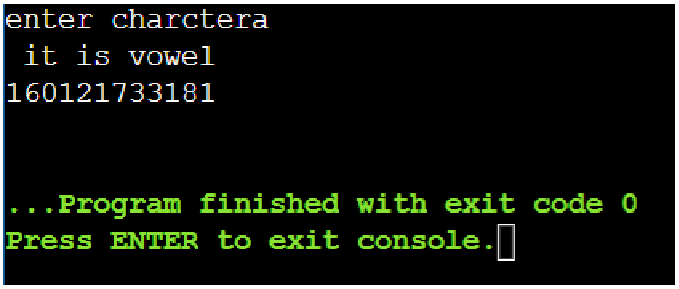
Aim: To determine whether the character entered is vowel or not.

Description: Here we are inputting a character and **Output**ting whether the entered character is vowel or not.

Code:

```
c=input("enter charcter")
if(c=="a" or c=="e" or c=="i" or c=="o" or c=="u"):
    print(" it is vowel")
elif(c=="A" or c=="E" or c=="I" or c=="O" or c=="U"):
    print("it is vowel")
else:
    print("not vowel")
print('160121733181')
```

Output:



```
enter charctera
 it is vowel
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

<u>Question 48:-</u>

Write a program to find the greatest number from three numbers

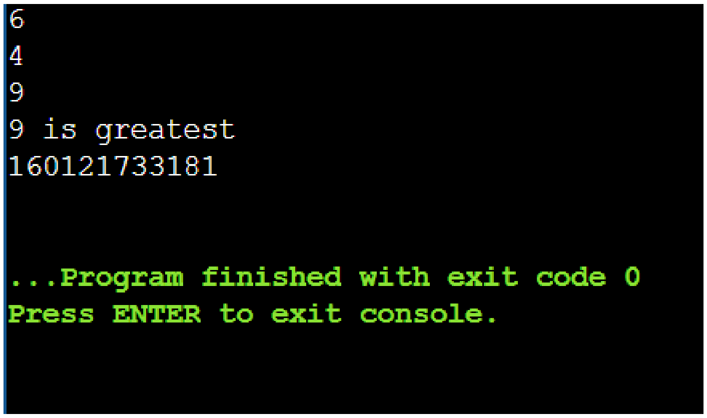
Aim: To find the greatest of three numbers

Description:In this program we append all the elements into a list and find the max from the list

<u>Code</u>:

```
x=int(input())
y=int(input())
z=int(input())
if (x>y and x>z):
   print(x,"is greatest")
elif(y>z and y>x):
   print(y,"is greatest")
else:
   print(z,"is greatest")
print('160121733181')
```

Output:



```
6
4
9
9 is greatest
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 49:-

Write a program that prompts the user to enter a number between 1-7 andthen displays the corresponding day of the week

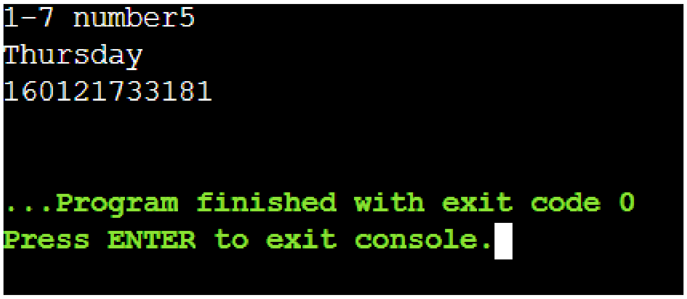
Aim: To prompt user to enter a number between 1-7 and display corresponding day of week

Description: In this program we take a input in between 1-7 and print the corresponding day of the week

Code:

```
n=int(input("1-7 number"))
if(n==1):
   print("Sunday")
elif(n==2):
   print("Monday")
elif(n==3):
   print("Tuesday")
elif(n==4):
   print("Wednesday")
elif(n==5):
   print("Thursday")
elif(n==6):
   print("Friday")
elif(n==7):
   print("Saturday")
print('160121733181')
```

Output:-



```
1-7 number5
Thursday
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 50:-

Write a program to calculate tax given the following conditions a. If incomeis less than1,50,000 then no tax b. If taxable income is 1,50,001-3,00,000 then charge 10% tax c. If taxable income is 3,00,001-5,00,000 then charge 20% tax d. If taxable income is above 5,00,001 then charge 30% tax

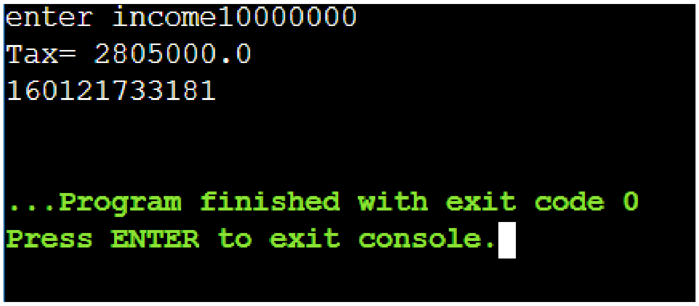
Aim: To calculate tax based on certain conditions

Description: This program tells that ,how much tax should a person pay the tax based on their income

Code:

```
income=int(input("enter income"))

ti=income-150000

tax=0

if (ti<150000):

    print("0 tax")

elif(ti>150000 and ti<=300000):

    tax=0.1*(ti-150000)

elif(ti>300000 and ti<=500000):

    tax=0.2*(ti-300000)

else:

    tax=0.3*(ti-500000)

print("Tax=",tax)

print('160121733181')
```

Output:



```
enter income10000000
Tax= 2805000.0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 51:-

Write a program to calculate the roots of a quadratic equation

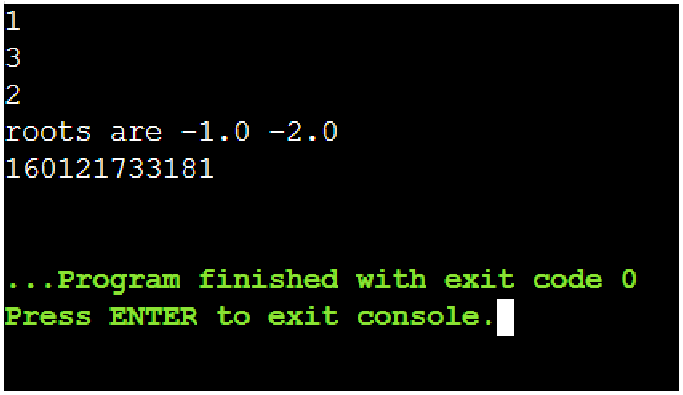
Aim: To calculate the roots of a quadratic equation

Description: In this program we basically find the roots of a quadratic eqn using roots formula.

Code:

```
import math
a=int(input())
b=int(input())
c=int(input())
d=math.sqrt(b*b-4*a*c)
if(d>0):
    r1=(-b+d)/(2*a)
    r2=(-b-d)/(2*a)
    print("roots are",r1,r2)
elif(d==0):
    r1=-b/(2*a)
    print("equal roots",r1,r1)
else:
    print("imaginary roots")
print('160121733181')
```

Output:



```
1
3
2
roots are -1.0 -2.0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 52 :

Write a program to read the numbers until -1 is encountered. Find the average of positive numbers and negative numbers entered by the user.

Aim: To read numbers until -1 is encountered and then find average of positive and negative numbers

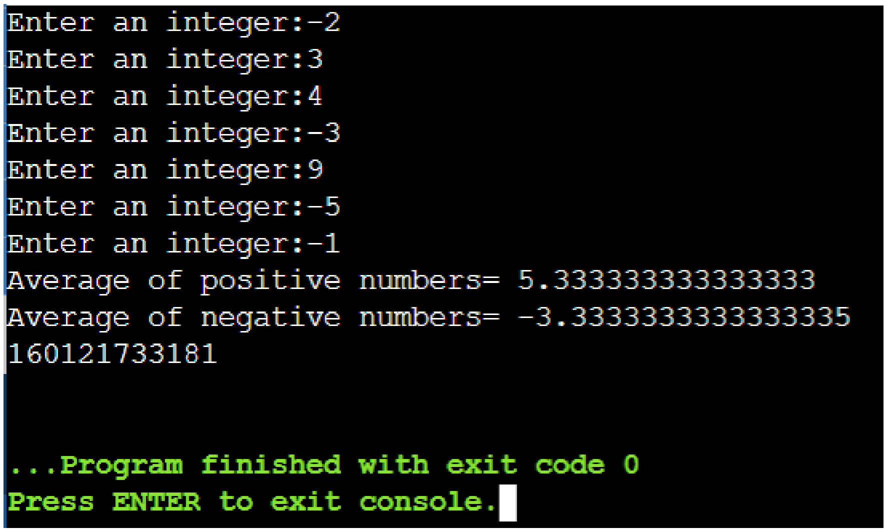
Description:

Here we construct a infinite loop and take inputs until we get an input ,finally we print the sum of positive and negative numbers

CODE :

```
num = int(input("Enter an integer:"))

p=[]

n=[]

while num!=-1 :

   if num>0:

      p.append(num)

   else :

      n.append(num)

   num = int(input("Enter an integer:"))

psum=0

nsum=0

for x in p :

   psum+=x

for x in n :

   nsum+=x

print("Average of positive numbers=",psum/len(p))

print("Average of negative numbers=",nsum/len(n))

print('160121733181')
```

Output :



```
Enter an integer:-2
Enter an integer:3
Enter an integer:4
Enter an integer:-3
Enter an integer:9
Enter an integer:-5
Enter an integer:-1
Average of positive numbers= 5.333333333333333
Average of negative numbers= -3.3333333333333335
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 53:-

Write a program to calculate the sum of numbers from m to n

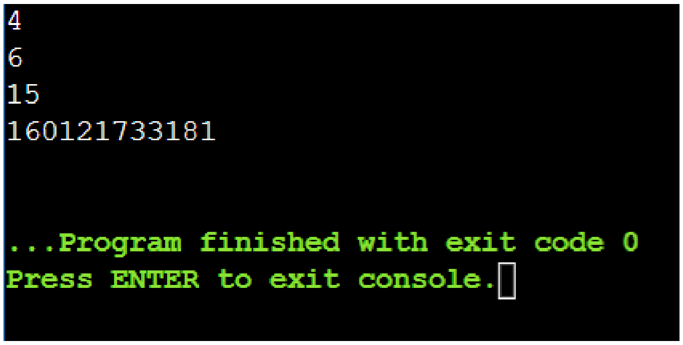
Aim: To calculate the sum of numbers from m to n

Description: We take m,n as inputs and find the sum of all numbers from m to n

Code:

```
m=int(input())

n=int(input())

sum=int(0)

for i in range(m,n+1):

    sum=sum+i

print(sum)

print('160121733181')
```

Output:



```
4
6
15
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 54 :

Write a program to read a character until * is encountered. Also count thenumber of uppercase, lowercase, and numbers entered by the users

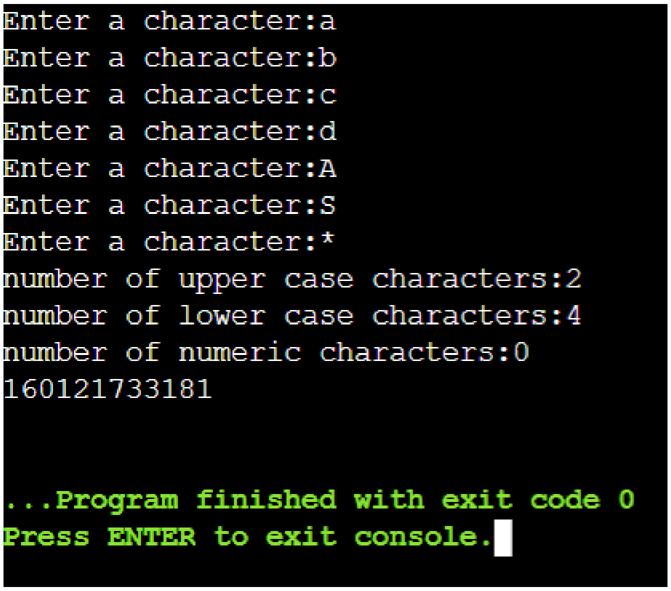
Aim: To read a character until * is encountered and then count number of upper, lower case alphabets and numbers entered

Description: In the above program we need to calculate the number of uppercase and lowercase letters and numbers entered by the user and print them when ‘*’ character is entered.

Code:

```
char = input("Enter a character:")
upperCount=0
lowerCount=0
numericCount=0
while char!='*' :
    if char.islower() :
        lowerCount+=1
    elif char.isupper() :
        upperCount+=1
    elif char.isdigit() :
        numericCount+=1
    else:
        print("Entered character is not an alphabet or a number")
    char = input("Enter a character:")
result = '''number of upper case characters:{}
number of lower case characters:{}
number of numeric characters:{}'''
print(result.format(upperCount,lowerCount,numericCount))
print('160121733181')
```

OUTPUT :



```
Enter a character:a
Enter a character:b
Enter a character:c
Enter a character:d
Enter a character:A
Enter a character:S
Enter a character:*
number of upper case characters:2
number of lower case characters:4
number of numeric characters:0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 55 :

Write a program using a while loop to read the numbers until -1 is encountered. Also, count the number of prime numbers and composite numbers entered by the user

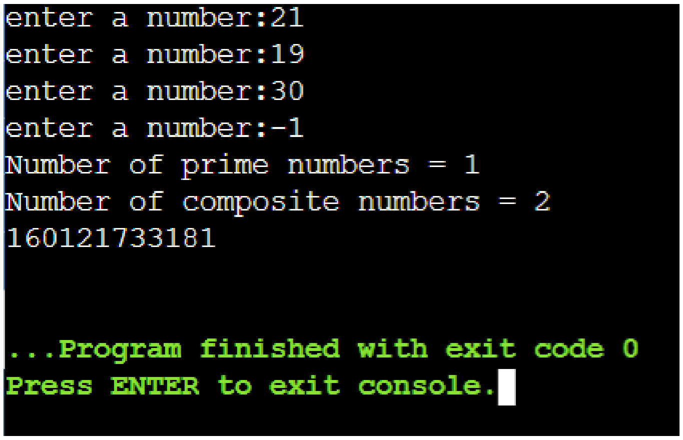
Aim: To read numbers until -1 is encountered and then count number of prime and composite numbers entered

Description: In the above program we need to calculate the number of prime and composite entered by the user and print the count of them when '-1' character is entered we need to stop the loop

CODE :

```
x = int(input("enter a number:"))

primeCount=0

compCount=0

while x!=-1 :

    for i in range(2,x) :

        if x%i==0 :

            compCount+=1

            break

    else :

        primeCount+=1

    x = int(input("enter a number:"))

print("Number of prime numbers =",primeCount)

print("Number of composite numbers =",compCount)

print('160121733181')
```

OUTPUT :



```
enter a number:21
enter a number:19
enter a number:30
enter a number:-1
Number of prime numbers = 1
Number of composite numbers = 2
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 56:-

Write a Program to sum the series 1+1/2+….+1/n

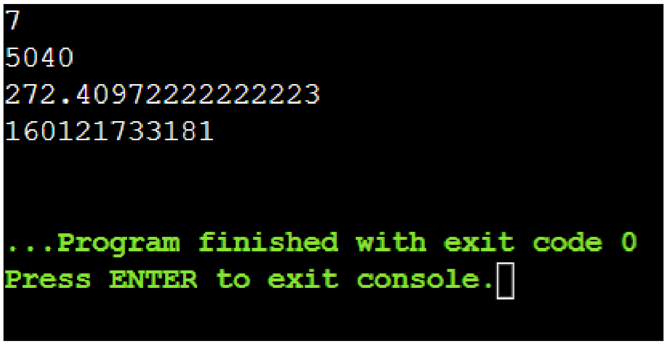
Aim: To sum the series 1+1/2+1/3+…1/n

Description: To find the sum of the elements in the given series

Code:

```
sum=0
n=int(input())
def fact(n):
    if n==1:
        return 1
    else:
        return n*fact(n-1)
def series(n):
    sum=0
    for i in range(1,n+1):
        sum+=(i**i)/fact(i)
    print(sum)
print(fact(n))
series(n)
print('160121733181')
```

Output:



```
7
5040
272.40972222222223
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 57:-

Write a python function to check whether three given numbers can form thesides of a triangle. Hint: Three numbers can be the sides of a triangle if none of the numbers are greater than or equal to the sum of the other two numbers.

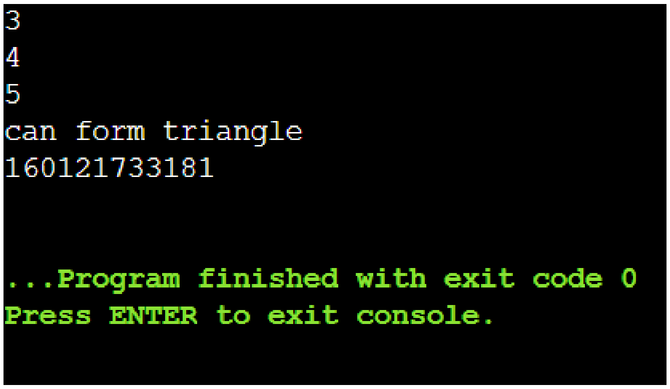
Aim: To check whether 3 given numbers can form the sides of a triangle.

Description: In the above program we need to check if the formation of a triangle is possible with the given dimensions of the sides.

Code:

```
def triangle(a,b,c):
    if a+b>=c and b+c>=a and c+a>=b:
        print("can form triangle")
    else:
        print("not possible")
a=int(input())
b=int(input())
c=int(input())
triangle(a,b,c)
print('160121733181')
```

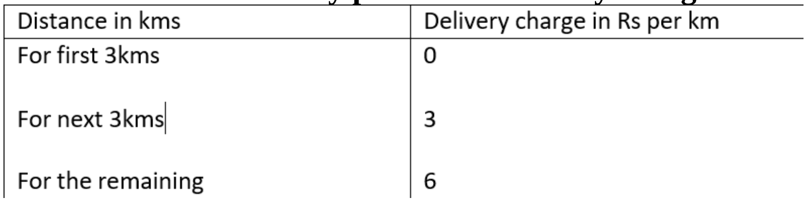
Output:



```
3
4
5
can form triangle
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

<u>Question 58:-</u>

FoodCorner home delivers vegetarian and non-vegetarian combos to its customer based on order. A vegetarian combo costs Rs.120 per plate and a non-vegetarian combo costs Rs.150 per plate. Their non-veg combo is reallyfamous that they get more orders for their non-vegetarian combo than the vegetarian combo. Apart from the cost per plate of food, customers are alsocharged for home delivery based on the distance in kms from the restaurantto the delivery point. The delivery charges are as mentioned below:



Distance in kms	Delivery charge in Rs per km
For first 3kms	0
For next 3kms	3
For the remaining	6

Given the type of food, quantity (no. of plates) and the distance in kms from the restaurant to the delivery point, write a python program to calculate the final bill amount to be paid by a customer. The below information must be used to check the validity of the data provided by the customer: Type of food must be ‘V’ for vegetarian and ‘N’ for non-vegetarian. Distance in kms must be greater than 0. Quantity ordered should be minimum 1. If any of theinput is invalid, the bill amount should be considered as -1.

Aim: To calculate price of bill based on type of food and distance

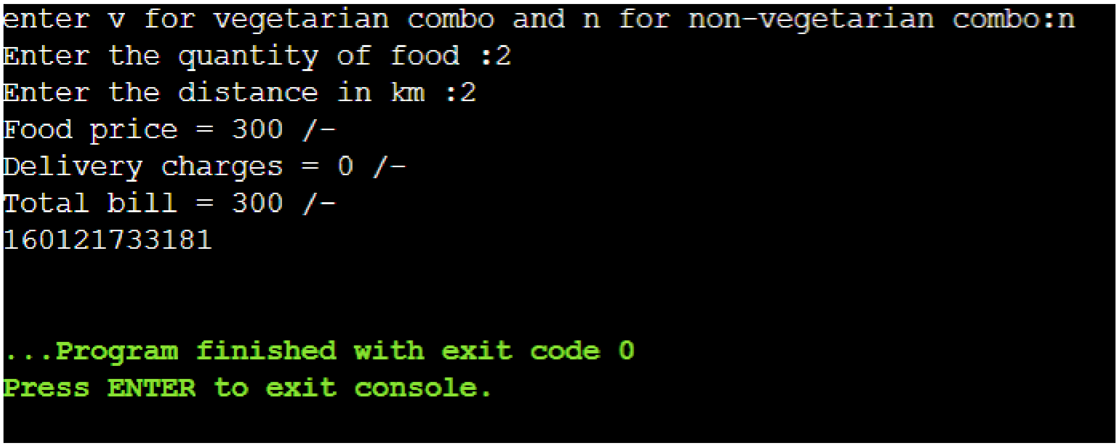
Description:

In the above program we need to calculate the bill by taking in the number of veg and nonveg combos from a given restaurant and print the bill.

<u>Code</u>:

```
food = input('enter v for vegetarian combo and n for non-vegetarian combo:')

quantity = int(input('Enter the quantity of food :'))

distance = int(input('Enter the distance in km :'))

def bill(food,quantity,distance) :

    if food=='v' :

        food_price=quantity*120

    elif food=='n' :

        food_price=quantity*150

    if distance<=3 :
        delivery_charges=0
    elif distance<=6 :
        delivery_charges=(distance-3)*3
    else :
        delivery_charges=(distance-6)*3+9
    print("Food price =",food_price,"/-")
    print("Delivery charges =",delivery_charges,"/-")
    print("Total bill =",food_price+delivery_charges,"/-")
if distance>0 and quantity>=1 :
    bill(food,quantity,distance)
else :
    print('-1')
print('160121733181')
```

OUTPUT :

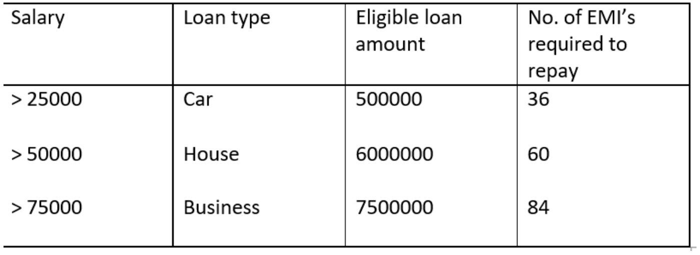


```
enter v for vegetarian combo and n for non-vegetarian combo:n
Enter the quantity of food :2
Enter the distance in km :2
Food price = 300 /-
Delivery charges = 0 /-
Total bill = 300 /-
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 59:-

The Metro Bank provides various types of loans such as car loans, businessloans and house loans to its account holders. Write a python program to implement the following requirements:

- **Initialize the following variables with appropriate input values:account_number, account_balance, salary, loan_type, loan_amount_expected and customer_emi_expected.**
- **The account number should be of 4 digits and its first digit should be 1.**
- **The customer should have a minimum balance of Rupees 1 Lakh in the account.**
- **If the above rules are valid, determine the eligible loan amount and the EMI that the bank can provide to its customers based on their salary and theloan type they expect to avail.**
- **The bank would provide the loan, only if the loan amount and the numberof EMI's requested by the customer is less than or equal to the loan amountand the number of EMI's decided by the bank respectively. Display appropriate error messages for all invalid data. If all the business rules are satisfied ,then display account number, eligible and requested loan amount and EMI's. Test your Code by providing different values for the input variables.**



Salary	Loan type	Eligible loan amount	No. of EMI's required to repay
> 25000	Car	500000	36
> 50000	House	6000000	60
> 75000	Business	7500000	84

Aim: To determine the loan and EMI to be given based on salary and minimum balance

Description:

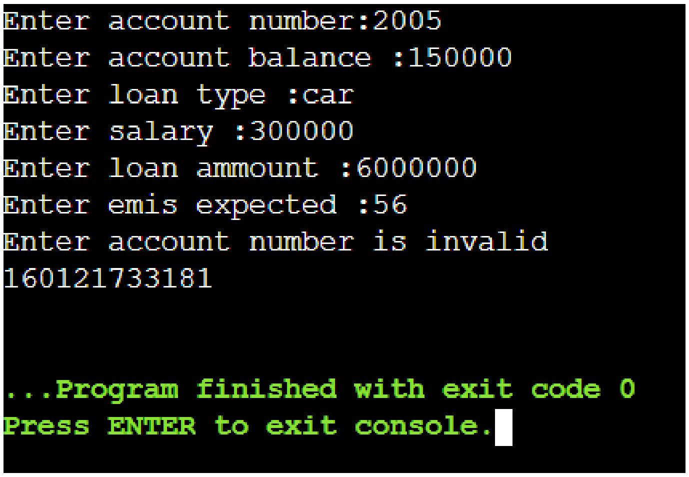
In the above program we need to calculate the bill by taking in the number of veg and nonveg combos from a given restaurant and print the bill.

Code:

```
acc_num=input('Enter account number:')

acc_bal=int(input('Enter account balance :'))

loan_type=input('Enter loan type :')
```

```
sal=int(input('Enter salary :'))
loan_amm=int(input('Enter loan ammount :' ))
emi=int(input('Enter emis expected :'))
def eligibility(sal,loan_type,loan_amm,emi,acc_num) :
    if loan_type=='Car' :
        if sal>=25000 :
            if loan_amm<=5000000 :
                if emi<=36 :
                    print(acc_num,'is eligible for a ',loan_type,'of loan ammount',loan_amm,'to repay with in ',emi,"emi's")
                else :
                    print("max emi's are 36 only")
            else :
                print("max loan ammount  is 5000000")
        else :
            print("minimum salary must be 25000 per month")
    elif loan_type=='House' :
        if sal>=50000 :
            if loan_amm<=6000000 :
                if emi<=60 :
                    print(acc_num,'is eligible for a ',loan_type,'of loan ammount',loan_amm,'to repay with in ',emi,"emi's")
                else :
                    print("max emi's are 60 only")
            else :
                print("max loan ammount  is 6000000")
        else :
            print("minimum salary must be 50000 per month")
    elif loan_type=='Business' :
        if sal>=75000 :
            if loan_amm<=7500000 :
                if emi<=84 :
                    print(acc_num,'is eligible for a ',loan_type,'of loan ammount',loan_amm,'to repay with in ',emi,"emi's")
                else :
                    print("max emi's are 84 only")
            else :
                print("max loan ammount  is 7500000")
        else :
            print("minimum salary must be 75000 per month")
if len(acc_num)==4 and acc_num[0]=='1' :
    if acc_bal>=100000 :
    eligibility(sal,loan_type,loan_amm,emi,acc_num)
    else:
    print('min acc balance is 100000')
else :
    print("Enter account number is invalid")
```

Output:



```
Enter account number:2005
Enter account balance :150000
Enter loan type :car
Enter salary :300000
Enter loan ammount :6000000
Enter emis expected :56
Enter account number is invalid
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 60:-

Write a python program to solve a classic ancient Chinese puzzle. We count35 heads and 94 legs among the chickens and rabbits in a farm. How many rabbits and how many chickens do we have?

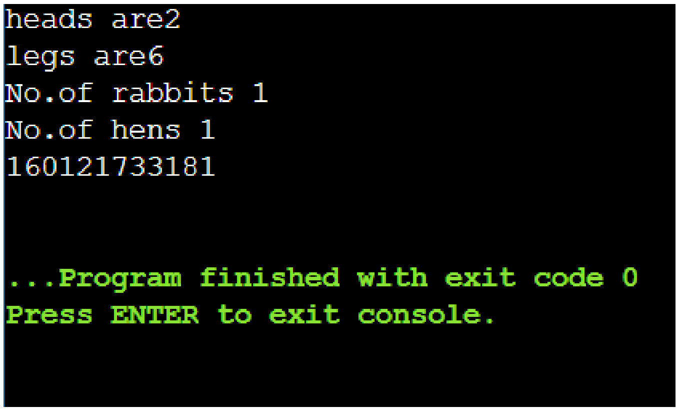
Aim: To count the number of chickens and rabbits based on number of legs and heads

Description: In the above program we need to calculate the number of rabbits and chickens from the number of heads and legs as the input.

Code:

```
def head_and_leg(heads,legs):
    if heads>=legs:
        print("NO solution")
    elif legs%2!=0:
        print("No solution")
    else:
        rabbit_count=(legs-2*heads)//2
        chicken_count=heads-rabbit_count
    print("No.of rabbits",rabbit_count)
    print("No.of hens",chicken_count)
heads=int(input("heads are"))
legs=int(input("legs are"))
head_and_leg(heads,legs)
print('160121733181')
```

Output:



```
heads are2
legs are6
No.of rabbits 1
No.of hens 1
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 61:-

Write a python program which finds the maximum number from num1 to num2 (num2 inclusive) based on the following rules. a. Always num1 should be less than num2 b. Consider each number from num1 to num2 (num2 inclusive). Populate the number into a list, if the below conditions are satisfied a. Sum of the digits of the number is a multiple of 3 b. Number has only two digits c. Number is a multiple of 5 c. Display the maximum element from the list In case of any invalid data or if the list is empty, display -1.

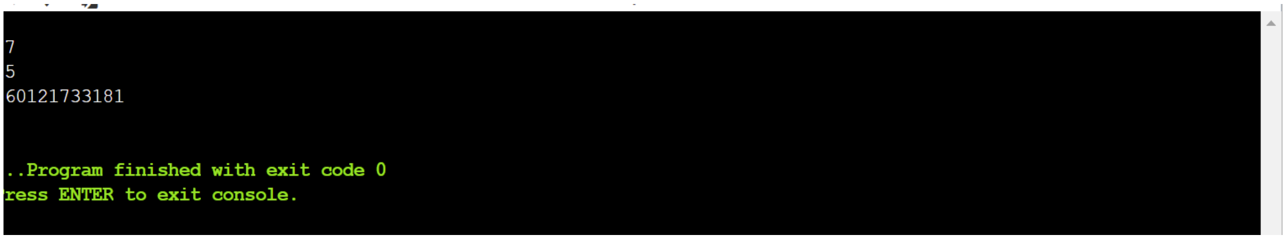
Aim: To find the maximum number from one number to another based on certain conditions

Description: In the above program we need to find the maximum number from a given range which is divisible by 3, 5 and has only 2 digits.

Code:

```
def digit_sum(n):
   s=0
   while n!=0:
      rem=n%10
      s+=rem
      n//=10
   return s
def digit_count(n):
   c=0
   while n!=0:
      n//=10
      c+=1
   return c
def maximum(num1, num2):
   if num2<num1:
      print("Num2 is lesser than Num1")
   else:
    l=[]
    for i in range(num1, num2+1):
        s=digit_sum(i)
        c=digit_count(i)
        if s%3==0 and c==2 and i%5==0:
            l.append(i)
    print(max(l))
num1=int(input())
num2=int(input())
maximum(num1, num2)
print("160121733181")
```

Output:



```
7
5
60121733181


..Program finished with exit code 0
ress ENTER to exit console.
```

Question 62:-

The flight ticket rates for a round-trip (Mumbai->Dubai) were as follows: Rate per Adult: Rs. 37550.0 Rate per Child: 1/3rd of the rate per adult Service Tax: 7% of the ticket amount (including all passengers) As it was a holiday season, the airline also offered 10% discount on the final ticket cost (after inclusion of the service tax). Find and display the total ticket cost for a group which had adults and children. Test the program with different input values for number of adults and children

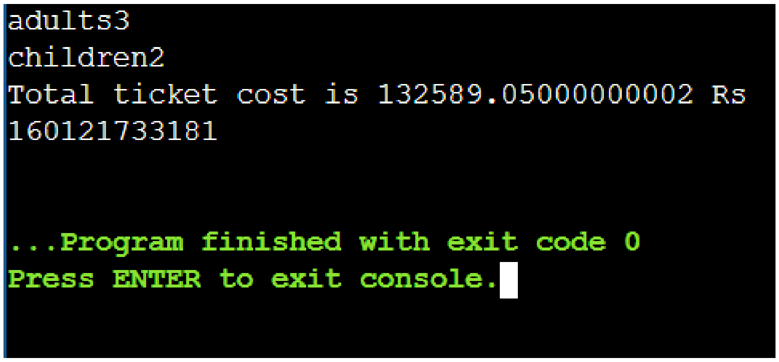
Aim: To calculate and display the total cost of airline tickets based on a group of adults and children

Description: In the above program we need to calculate the total fare for an air ticket for a family by taking the number of adults and children.

Code:

```
def ticketcost(a,c):
   t=(a*37550)+(c*37550/3)
   t+=(t*0.07)
   t-=(t*0.1)
   print("Total ticket cost is",t,"Rs")
a=int(input("adults"))
c=int(input("children"))
ticketcost(a,c)
print('160121733181')
```

Output:



```
adults3
children2
Total ticket cost is 132589.05000000002 Rs
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 63:-

Given a list of integer values. Write a python program to check whether it contains same number in adjacent position. Display the count of such adjacent occurrences

Aim: To check whether a list of integers contains same number in adjacent position and print number of such occurrences

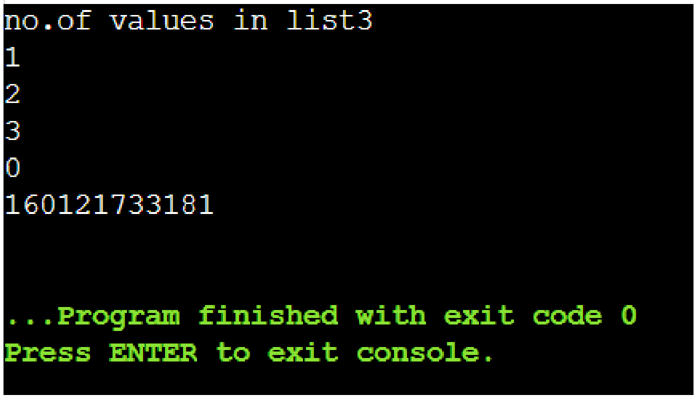
Description:

In the above program we need to count the number of adjacent occurrences of a number in the list. This can be done by checking two adjacent elements and increasing the count of the variable containing the number of such occurrences. We need to use one for loop to check the adjacent numbers and use a conditional statement.

Code:

```
n=int(input("no.of values in list"))
def counter(l):
    count=0
    for i in range(n-1):
        if l[i]==l[i+1]:
            count+=1
        else:
            continue
    return count
l=[]
for i in range(n):
    l.append(int(input()))
print(counter(l))
counter(l)
print('160121733181')
```

Output:



```
no.of values in list3
1
2
3
0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 64:-

Write a Python program to generate the next 15 leap years starting from a given year. Populate the leap years into a list and display the list

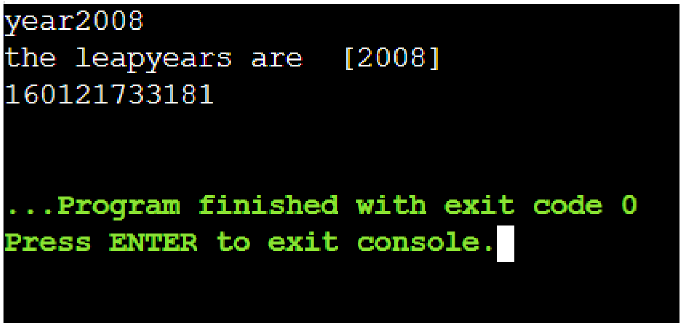
Aim: To generate the next 15 leap years starting from a given year

Description: In the above program we need to print the next 15 leap years given the year input. To accomplish this task we can use a for loop for 15 times and using conditional statements we can append the leap years in a list and print the list after the elements are completely updated.

Code:

```
def leap_years(year):
    count=0
    list=[]
    while(count<15):
        if(year%4==0 or year%400==0 and year%100==0):
            list.append(year)
            count=count+1
            year+=1
        return list
year=int(input("year"))
list=leap_years(year)
print("the leapyears are ",list)
leap_years(year)
print('160121733181')
```

Output:



```
year2008
the leapyears are  [2008]
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

Question 65:-

ARS Gems Store sells different varieties of gems to its customers. Write a Python program to calculate the bill amount to be paid by a customer based on the list of gems and quantity purchased. Any purchase with a total bill amount above Rs.30000 is entitled for 5% discount. If any gem required by the customer is not available in the store, then consider total bill amount to be -1. Assume that quantity required by the customer for any gem will always be greater than 0. Perform case-sensitive comparison wherever applicable

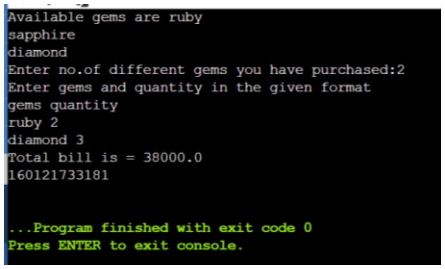
Aim: To find the bill amount of purchasing different types of gems

Description: In this program we first ask user to input what are the different items he is purchasing then asking quantity of each item and calculating bill based on given conditions we calculate the bill aand their by we print it.

Code:

```
print("Available gems are ruby\nsapphire\ndiamond")
n=int(input("Enter no.of different gems you have purchased:"))
def gem(n):
    print("Enter gems and quantity in the given format\ngems quantity")
    bill=0
    for i in range(n):
        x,y=input().split()
        y=int(y)
        if x.lower()=="ruby":
            cost=y*5000
        elif x.lower()=="sapphire":
            cost=y*7000
        elif x.lower()=="diamond":
            cost=y*10000
        else:
            bill=-1
            print("Total bill =",bill)
        bill+=cost
    if(bill>30000):
        bill=bill-0.05*bill
        print("Total bill is =",bill)
    else:
        print("Total bill =",bill)
gem(n)
print('160121733181')
```

Output:



```
Available gems are ruby
sapphire
diamond
Enter no.of different gems you have purchased:2
Enter gems and quantity in the given format
gems quantity
ruby 2
diamond 3
Total bill is = 38000.0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 66 :

Write a python function, create_largest_number(), which accepts a list of numbers and returns the largest number possible by concatenating the list of numbers. Note: Assume that all the numbers are two digit numbers

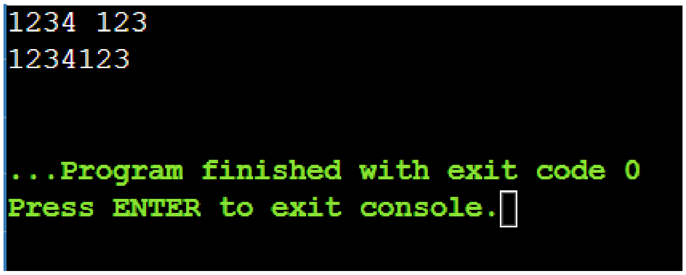
Aim: To accept a list of numbers and return largest number by concatenating the list

Description: In this above program we need to print the highest possible number that can be formed from the input numbers. This can be done by first appending the elements in a list and then sorting it in the descending order and then concatenating the elements in an empty list and then printing the string.

Code:

```
nums = list(map(int,input().split()))

nums.sort(reverse=True)

res="

for x in nums :

   res+=str(x)

print(res)
```

Output:



```
1234 123
1234123


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 67 :
WAP to find sum of series 1/1! + 4/2! + 27/3! + … using functions

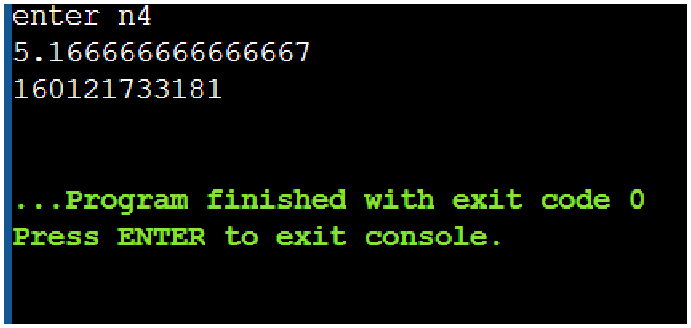
Aim: To find the sum of series using functions

Description: We need to find the sum of series provided here ,with the help of recursion we can construct a function then we can stop at that func at a particular base case and finally **Output** the sum

Code:

```
sum=0

n=int(input("enter n"))

def fact(n):

    if n==1:

        return 1

    else:

        return n*fact(n-1)

def series(n):

    sum=0

    for i in range(1,n+1):

        sum+=(i*i)/fact(i)

    print(sum)

fact(n)

series(n)

print('160121733181')
```

Output:



```
enter n4
5.166666666666667
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 68 :
WAP to display powers of 2 using Anonymous function or lambda function

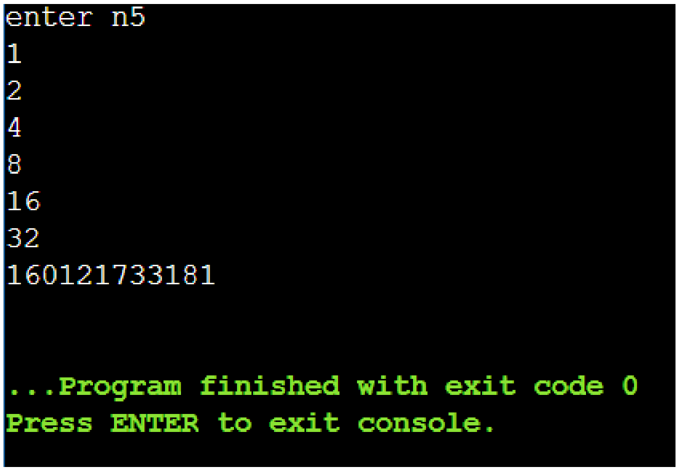
Aim: To display powers of 2 using lambda function

Description:Creating a lambda function which gives power of 2 upto certain value.

Code:

```
def lambdafunction(n):
   for i in range(0,n+1):
      x=lambda x:2**x
      print(x(i))
n=int(input("enter n"))
lambdafunction(n)
print('160121733181')
```

Output:



```
enter n5
1
2
4
8
16
32
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 69 :

Write a Python program to find factorial of a given number using recursive lambda function

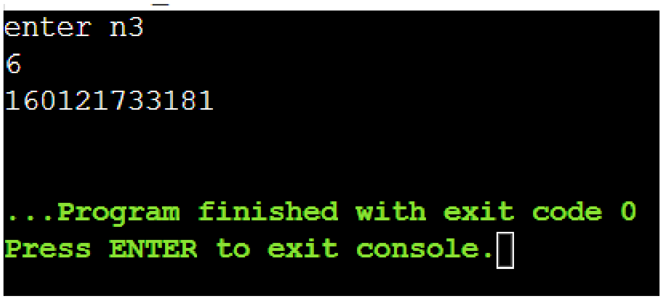
Aim: To find the factorial of a given number using lambda function

Description: We are constructing a recursive lambda function such that it gives factorial of a given number

<u>Code</u>:

```
x=lambda x:x*fact(x-1)

def fact(n):

   if n==1:

      return 1

   else:

      return x(n)

n=int(input("enter n"))

print(fact(n))

fact(n)

print('160121733181')
```

Output:



```
enter n3
6
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 70 :
WAP to illustrate command line arguments and function redefinition

Aim: To illustrate command line arguments and function re definition

Description: In this program we discuss the usage of command line arguments and function redefinition

CODE :

```
import sys

a=int(sys.argv[1])

b=int(sys.argv[2])

print(a+b)

def hi() :

   print('hello')

hello=hi

hi()

hello()
```

OUTPUT :

QUESTION 71 :
WAP that has a class Person storing name and date of birth of a person. The program should subtract the DOB from today's date to find out whether a person is eligible to vote or not

Aim: To store name and date of birth of a person and find out whether the person is eligible to vote

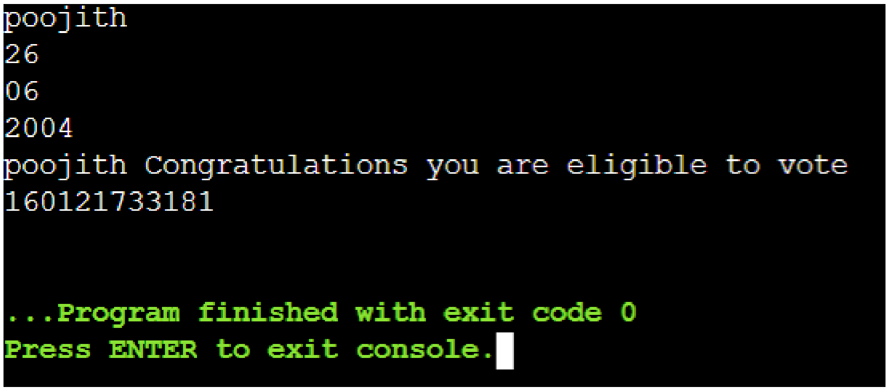
Description:creating a class having DOB of a person and finding his age and finding whether he is eligible for voting or not.

Code:

```
import datetime
class person:
   def __init__(self,name,dob):
      self.name=name
      self.dob=dob
   def check(self):
      today=datetime.date.today()
      age=today.year-self.dob.year
      if today<datetime.date(today.year,self.dob.month,self.dob.day):
         age-+1
      if age>=18:
         print(self.name,"Congratulations you are eligible to vote")
      else:
         print(self.name,"sorry")
name=input()
d=int(input())
m=int(input())
y=int(input())
p=person(name,datetime.date(y,m,d))
p.check()
```

print('160121733181')

Output:



```
poojith
26
06
2004
poojith Congratulations you are eligible to vote
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 72 :
WAP that has a class Student that stores roll, name and marks(3 subjects) of the students. Display the information of the student with his/her percentage

Aim: To make a class to store name, roll number and marks and display the percentage of marks

Description: Creating a class having marks of 3 subjects and generating a function which tells percentage of the respective student.

<u>Code</u>:

```
class student:
   marks=[]
   def Marks(self,rn,name,m1,m2,m3):
      student.rn=rn
      student.name=name
      student.marks.append(m1)
      student.marks.append(m2)
      student.marks.append(m3)
   def display(self):
      print("Roll no is",student.rn)
      print("Name is",student.name)
      print("Marks are",student.marks)
      print("average marks are",self.average())
   def average(self):
      return  ((student.marks[0]+student.marks[1]+student.marks[2])/3)
rn=int(input("roll"))
name=input("name")
m1=int(input("marks 1"))
m2=int(input("marks 2"))
m3=int(input("marks 3"))
```

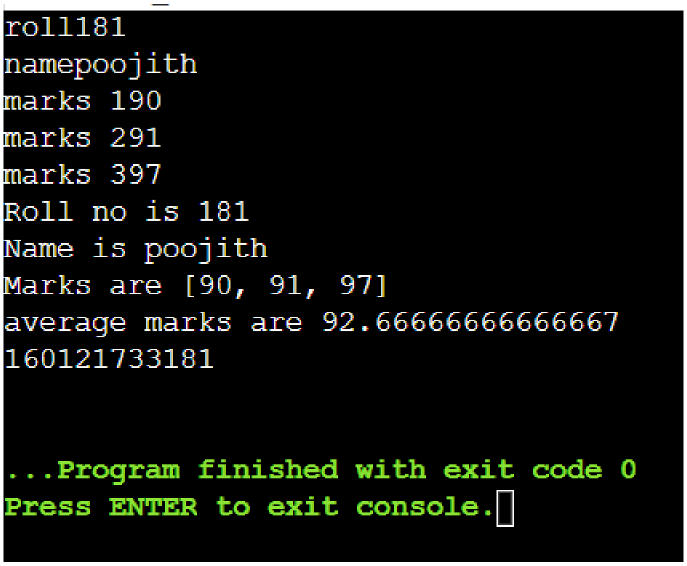
s1=student()

s1.Marks(rn,name,m1,m2,m3)

s1.display()

print('160121733181')

Output:



```
roll181
namepoojith
marks 190
marks 291
marks 397
Roll no is 181
Name is poojith
Marks are [90, 91, 97]
average marks are 92.66666666666667
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 73 :

Write a program that accepts the lengths of three sides of a triangle as inputs. The program Output should indicate whether or not the triangle is a right triangle (Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).

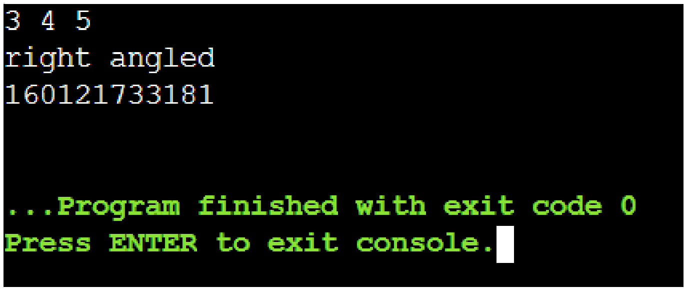
Aim: To accept lengths of sides of triangle and check whether it can form a right angled triangle

Description: We need to check if the input sides form a right angle triangle. We can verify this using the Pythagorean theorem.

Code:

```
from math import sqrt

a,b,c=map(int,input().split(' '))

if a>b and a>c:

   if a==sqrt(b**2+c**2):

      print("right angled")

   else:

      print("not right angled")

elif b>c and b>a:

   if b==sqrt(c**2+a**2):

      print("right angled")

   else:

      print("not right angled")

elif c>b and c>a:

   if c==sqrt(b**2+a**2):

      print("right angled")

   else:

      print("not right angled")

print('160121733181')
```

Output:



```
3 4 5
right angled
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 74 :

Write a python program to define a module to find Fibonacci Numbers and import the module to another program

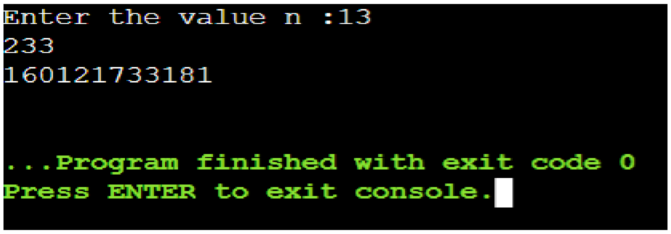
Aim: To define a module to find Fibonacci numbers and import module to another program

Description: In the above program we need to print the Fibonacci series by importing a user created module in another program.

CODE :

```
import fibbo

print(fibbo.fibbonacci(int(input('Enter the value n :'))))


#fibbo module

def fibbonacci(n) :

    c=0

    if n==1 or n==2 :

        c=1

    else :

        a=1

        b=1

        for i in range(3,n+1) :

            c=a+b

            a=b

            b=c

    return c
```

OUTPUT :



```
Enter the value n :13
233
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 75 :

Write a python program to define a module and import a specific function in that module to another program

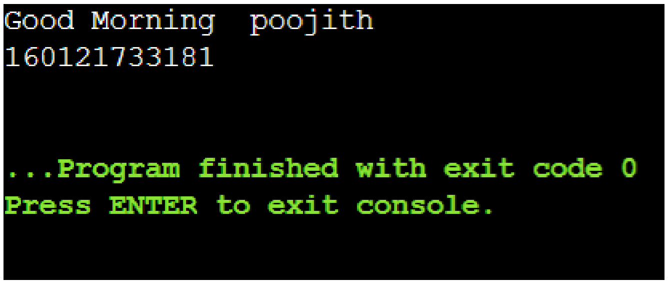
Aim: To define a module and import a specific function from it into another program

Description: In this program we create a module which has some functions in it and we import this module and use the functions in it

CODE :

```
from mod_1 import wish

wish('poojith')


#mod1 module

def print_hello(name) :

   print('hello! ',name)

def wish(name) :

   print('Good Morning ',name)
```

OUTPUT :



```
Good Morning  poojith
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 7 :

WAP that has a class Person storing name and date of birth of a person. The program should subtract the DOB from today's date to find out whether a person is eligible to vote or not.

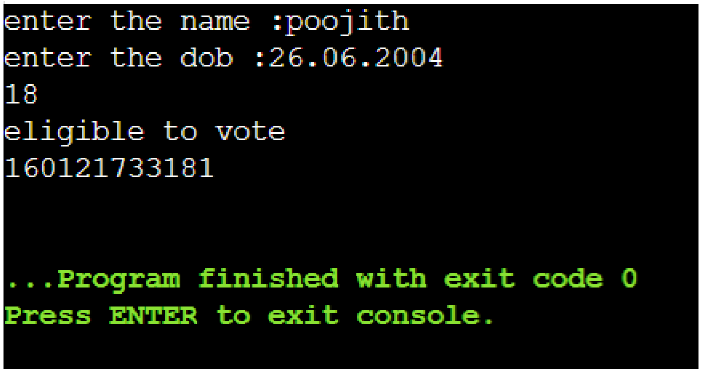
Aim: To store name and date of birth of a person and find out whether the person is eligible to vote

Description:creating a class having DOB of a person and finding his age and finding whether he is eligible for voting or not.

CODE :

```
from datetime import date
class person :
   def __init__(self,name,dob) :
      self.name=name
      self.dob=dob
   def vote(self) :
      dob=self.dob
      days_in_year = 365.2425
      age = int((date.today() - dob).days / days_in_year)
      print(age)
      if(age>=18) :
         print('eligible to vote')
      else :
         print('not eligible to vote')
name=input('enter the name :')
dob=input('enter the dob :').split('.')
p1=person(name,date(int(dob[2]),int(dob[1]),int(dob[0])))
p1.vote()
```

OUTPUT :



```
enter the name :poojith
enter the dob :26.06.2004
18
eligible to vote
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 77 :

WAP that has a class Student that stores roll, name and marks(3 subjects) of the students. Display the information of the student with his/her percentage

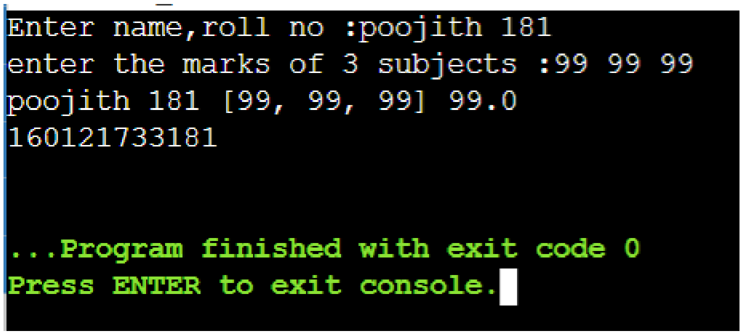
Aim: To make a class to store name, roll number and marks and display the percentage of marks

Description: Creating a class having marks of 3 subjects and generating a function which tells percentage of the respective student.

CODE :

```
class student :
    def __init__(self,roll,name,marks) -> None:
        self.roll=roll
        self.name=name
        self.marks=marks
        self.perc=(self.marks[0]+self.marks[1]+self.marks[2])/3
        print(self.name,self.roll,self.marks,self.perc)
x=input('Enter name,roll no :').split()
y=list(map(int,input('enter the marks of 3 subjects :').split()))
student(x[1],x[0],y)
print('160121733181')
```

OUTPUT :_



```
Enter name,roll no :poojith 181
enter the marks of 3 subjects :99 99 99
poojith 181 [99, 99, 99] 99.0
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 76 :

WAP that has a class Store which keeps a record of **Code** and price of each product. Display a menu of all products to the user and prompt him to enter the quantity of each item required. Generate a bill and display the total amount.

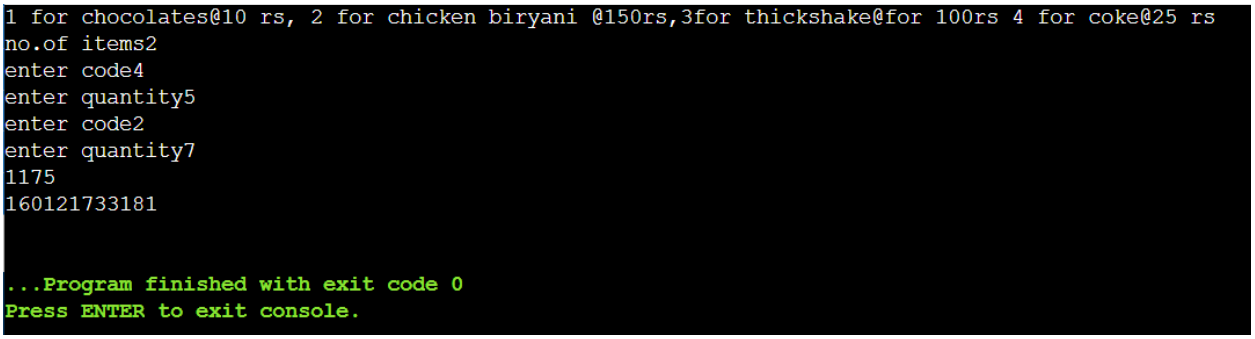
Aim: To create a class to store **Code** and price of each product and calculate a bill based on items bought by a user.

Description: In this program we need to create a class which stores the **Code** and price of each product in a store and print the bill according to the users input.

CODE :

```
class Store:
   def __init__(self,Code,quantity,price):
      self.Code=Code
      self.price=price
      self.quantity=quantity
   def display(self):
      amt=self.price*self.quantity
      l.append(amt)
      return sum(l)
l=[]
print("1 for chocolates@10 rs, 2 for chicken biryani @150rs,3for thickshake@for 100rs 4 for coke@25 rs")
for i in range(int(input("no.of items"))):
   a=int(input("enter Code"))
   b=int(input("enter quantity"))
   if a==1:
      price=10
   elif a==2:
        price=150
    elif a==3:
        price=100
    elif a==4:
        price=25
    p=Store(a,b,price)
    d=p.display()
print(d)
print('160121733181')
```

Output:



```
1 for chocolates@10 rs, 2 for chicken biryani @150rs,3for thickshake@for 100rs 4 for coke@25 rs
no.of items2
enter code4
enter quantity5
enter code2
enter quantity7
1175
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

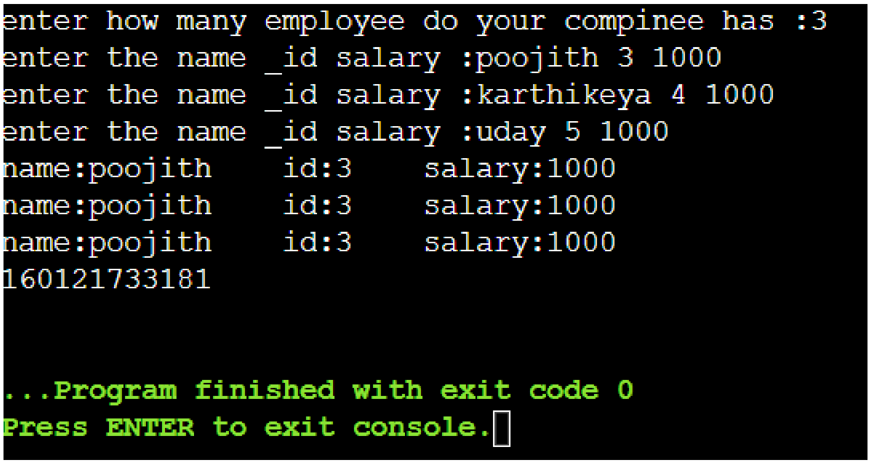
QUESTION 77 :

Define a class Employee. Display the personal and salary details of fiveemployees using single inheritance.

CODE :

```
class storing_val:
   def __init__(self,name,_id,salary):
       self.name=name
       self._id=_id
       self.salary=salary
class Employee(storing_val):#inheritance taking placE
   def display(self):
       print(f"name:{self.name}\tid:{self._id}\tsalary:{self.salary}")
store=[]
n=int(input("enter how many employee do your compinee has :"))
for i in range(n):
   x=input("enter the name _id salary :").split()
   store.append(x)
for i in range(n):
   #print("DETAILS OF ",i+1,"th employee")
   j=Employee(store[0][0],store[0][1],store[0][2])
   Employee.display(j)
```

OUTPUT :



```
enter how many employee do your compinee has :3
enter the name _id salary :poojith 3 1000
enter the name _id salary :karthikeya 4 1000
enter the name _id salary :uday 5 1000
name:poojith    id:3    salary:1000
name:poojith    id:3    salary:1000
name:poojith    id:3    salary:1000
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

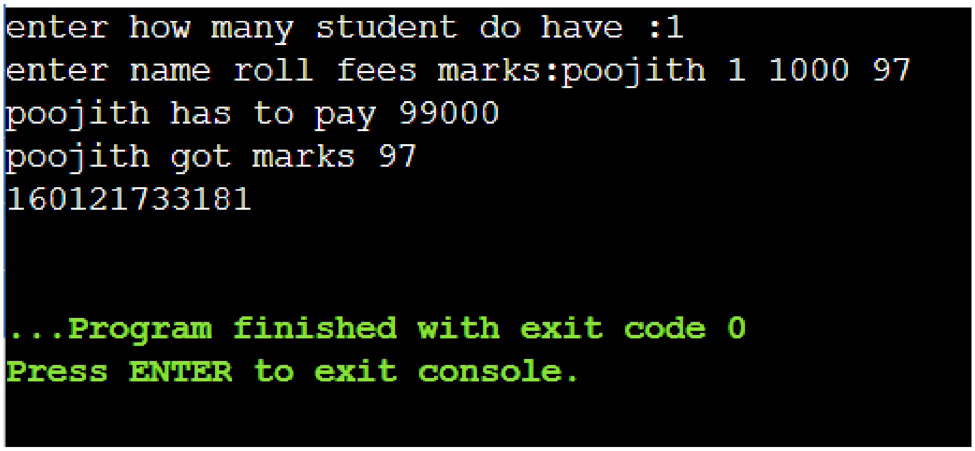
QUESTION 78 :

Define a class student with data members as roll no and name. Derive a classFees from student that has data member fees and functions to submit fees andgenerate receipt. Derive another class result from student that displays the Marks and grade obtained by the student.

CODE :

```
class student:
   def __init__(self,name,roll):
      self.name=name
      self.roll=roll
class fee(student):
   def __init__(self,name,roll,f):
      self.f=f
      super().__init__(name,roll)
   def fee_check(self):
      return (100000-self.f)
class marks(student):
   def __init__(self,name,roll,m):
      self.marks=m
      super().__init__(name,roll)
   def marks_display(self):
      return self.marks
store=[]
n=int(input("enter how many student do have :"))
for  i in range(n):
   x=input("enter name roll fees marks:" ).split()
   store.append(x)
for i in range(n):
   r=int(store[i][1])
   fe=int(store[i][2])
   ma=int(store[i][3])
   j=fee(store[i][0],r,fe)
   print(f"{store[i][0]} has to pay {j.fee_check()}")
   j=marks(store[i][0],r,ma)
   print(f"{store[i][0]} got marks {j.marks_display()}")
```

OUTPUT :



```
enter how many student do have :1
enter name roll fees marks:poojith 1 1000 97
poojith has to pay 99000
poojith got marks 97
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 79 :

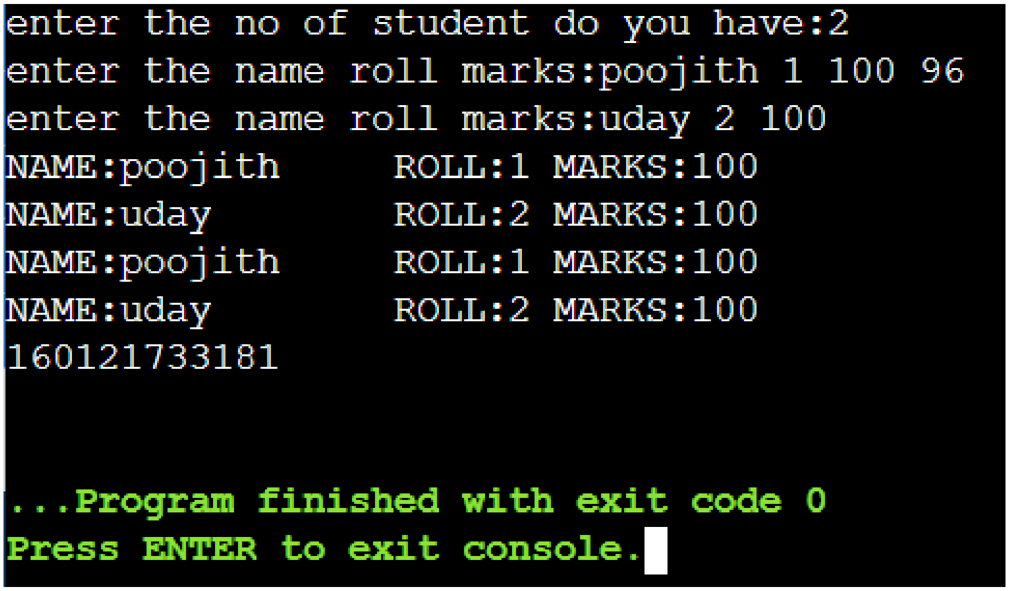
Write a program that has a class student to store the details of students in a class. Derive another class toppers from the student that stores records of onlytop 3 students of the class.

CODE :

```
class student:
   l=[]
   def __init__(self,name,roll,marks):
      self.name=name
      self.roll=roll
      self.marks=marks
      student.l.append(self)
class topper(student):
   def display(self):
      print(f"NAME:{self.name}\t ROLL:{self.roll}\tMARKS:{self.marks}")

input_taker=[]
n=int(input("enter the no of student do you have:"))
for i in range(n):
   x=input("enter the name roll marks:").split()
   input_taker.append(x)
for i in range(n):
   j=int(input_taker[i][1])
   k=input_taker[i][0]
   m=int(input_taker[i][2])
   caller=student(k,j,m)
dt=student.l.copy()
marks_list=[]
for i in range(n):
   marks_list.append(dt[i].marks)
m=sorted(marks_list)
m=m[::-1]
m=m[0:3]
for i in m:
   for j in range(n):
      if(i==student.l[j].marks):
         kk=int(input_taker[j][1])
         k=input_taker[j][0]
```

```
        m=int(input_taker[j][2])
        caller1=topper(k,kk,m)
        caller1.display()
```

OUTPUT :



```
enter the no of student do you have:2
enter the name roll marks:poojith 1 100 96
enter the name roll marks:uday 2 100
NAME:poojith     ROLL:1 MARKS:100
NAME:uday        ROLL:2 MARKS:100
NAME:poojith     ROLL:1 MARKS:100
NAME:uday        ROLL:2 MARKS:100
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 80 :

Write a program that has a class train with data members no_of_ seats_1 st,no_of_ seats_2Tier, no_of_ seats_3Tier and member functions to set and display the data. Derive a class Reservation that has the data members seats_booked_1 st ,seats_booked_2 tier and the seats_booked_3 tier and functions to book and cancel tickets and display status.

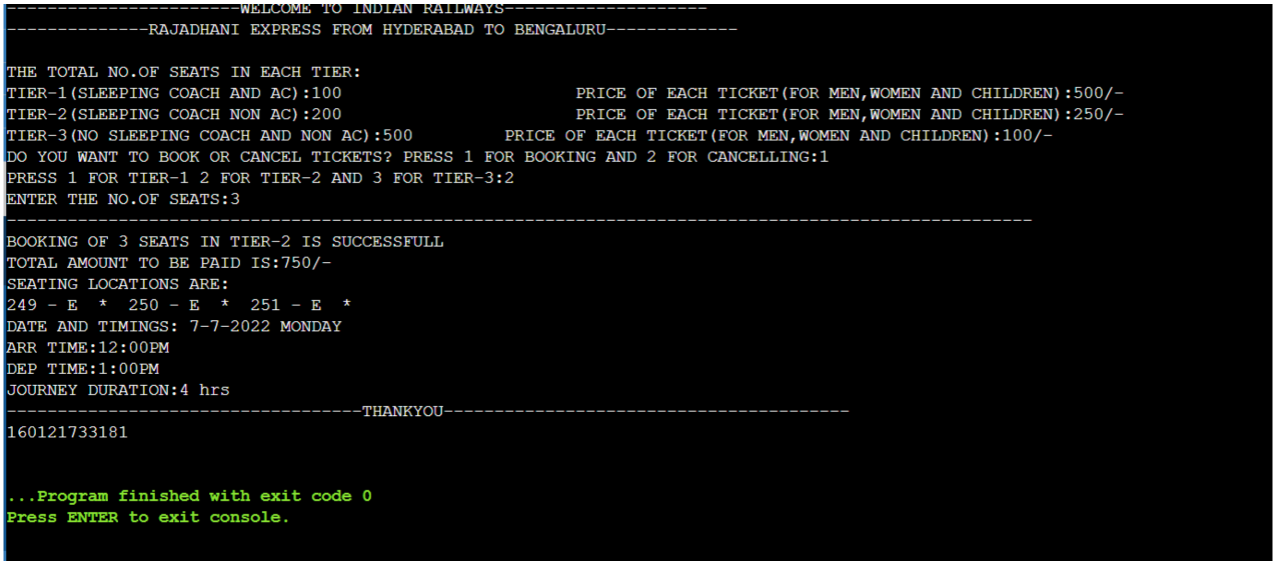
CODE :

```
from random import randint,choice
class train:
    def __init__(self):
        self.no_of_seats_1tier = 100
        self.no_of_seats_2tier = 200
        self.no_of_seats_3tier = 500
        self.ticket_price_1tier = 500
        self.ticket_price_2tier = 250
        self.ticket_price_3tier = 100
    def display(self):
        print("\nTHE TOTAL NO.OF SEATS IN EACH TIER: ")
        print(f"TIER-1(SLEEPING COACH AND AC):{self.no_of_seats_1tier}\t\t\tPRICE OF
EACH TICKET(FOR MEN,WOMEN AND CHILDREN):{self.ticket_price_1tier}/-")
        print(f"TIER-2(SLEEPING COACH NON AC):{self.no_of_seats_2tier}\t\t\tPRICE OF
EACH TICKET(FOR MEN,WOMEN AND CHILDREN):{self.ticket_price_2tier}/-")
        print(f"TIER-3(NO SLEEPING COACH AND NON AC):{self.no_of_seats_3tier}\t
PRICE OF EACH TICKET(FOR MEN,WOMEN AND CHILDREN):{self.ticket_price_3tier}/-
")
class reservation(train):
    def __init__(self,type,tier,no_of_seats):
        super().__init__()
        self.type = type
        self.no_of_seats = no_of_seats
        self.tier = tier
        self.seats_booked_1tier = randint(1,101)
        self.seats_booked_2tier = randint(1,201)
        self.seats_booked_3tier = randint(1,501)
        self.bill = 0
        self.refund = 0
    def booking(self):
        if self.type == 1:
            if self.tier == 1:
                if self.seats_booked_1tier + self.no_of_seats <= self.no_of_seats_1tier:
                      self.bill = self.ticket_price_1tier*self.no_of_seats
                      print("------------------------------------------------------------------------------------------
----------")
                      print(f"BOOKING OF {self.no_of_seats} SEATS IN TIER-1 IS
SUCCESSFULL")
                      print(f"TOTAL AMOUNT TO BE PAID IS:{self.bill}/-")
                      print("SEATING LOCATIONS ARE:")
                      CHOICE = choice(["A", "B", "C", "D", "E"])
                      SEAT_NO = randint(1, 101)
                      for i in range(self.no_of_seats):
                          print(SEAT_NO+i,"-",CHOICE,end = "  *  ")
               else:
                   print(f"SORRY ONLY {(self.seats_booked_1tier+self.no_of_seats)-100} ARE
LEFT!!")
           elif self.tier == 2:
               if self.seats_booked_2tier + self.no_of_seats <= self.no_of_seats_2tier:
                      self.bill = self.ticket_price_2tier*self.no_of_seats
                      print("------------------------------------------------------------------------------------------
----------")
                      print(f"BOOKING OF {self.no_of_seats} SEATS IN TIER-2 IS
SUCCESSFULL")
                      print(f"TOTAL AMOUNT TO BE PAID IS:{self.bill}/-")
                      print("SEATING LOCATIONS ARE:")
                      CHOICE = choice(["A", "B", "C", "D", "E"])
                      SEAT_NO = randint(100, 301)
                      for i in range(self.no_of_seats):
                          print((SEAT_NO+ i),"-",CHOICE, end="  *  ")
               else:
                   print(f"SORRY ONLY {(self.seats_booked_2tier+self.no_of_seats)-200} ARE
LEFT!!")

           elif self.tier == 3:
               if self.seats_booked_3tier + self.no_of_seats <= self.no_of_seats_3tier:
                      self.bill = self.ticket_price_3tier*self.no_of_seats
                      print("------------------------------------------------------------------------------------------
----------")
                      print(f"BOOKING OF {self.no_of_seats} SEATS IN TIER-3 IS
SUCCESSFULL")
                      print(f"TOTAL AMOUNT TO BE PAID IS:{self.bill}/-")
```

```
                        print("SEATING LOCATIONS ARE:")
                        CHOICE = choice(["A", "B", "C", "D", "E"])
                        SEAT_NO = randint(300, 801)
                        for i in range(self.no_of_seats):
                            print((SEAT_NO+ i),"-",CHOICE, end="  *   ")
                else:
                    print(f"SORRY ONLY {(self.seats_booked_3tier+self.no_of_seats)-500} ARE
LEFT!!")

            print("\nDATE AND TIMINGS: 7-7-2022 MONDAY \nARR TIME:12:00PM \nDEP
TIME:1:00PM \nJOURNEY DURATION:4 hrs")
        elif self.type ==2:
            if self.tier == 1:
                print(f"CANCELLATION OF {self.no_of_seats} SEATS IN TIER-1 IS
SUCCESSFULL")
                self.refund = self.ticket_price_1tier*self.no_of_seats
                print(f"TOTAL AMOUNT OF {self.refund}/- WILL BE REFUNDED SHORTLY")
            elif self.tier == 2:
                print(f"CANCELLATION OF {self.no_of_seats} SEATS IN TIER-2 IS
SUCCESSFULL")
                self.refund = self.ticket_price_2tier*self.no_of_seats
                print(f"TOTAL AMOUNT OF {self.refund}/- WILL BE REFUNDED SHORTLY")
            elif self.tier == 3:
                print(f"CANCELLATION OF {self.no_of_seats} SEATS IN TIER-3 IS
SUCCESSFULL")
                self.refund = self.ticket_price_3tier*self.no_of_seats
                print(f"TOTAL AMOUNT OF {self.refund}/- WILL BE REFUNDED SHORTLY")
#inputting
print("-----------------------WELCOME TO INDIAN RAILWAYS--------------------")
print("--------------RAJADHANI EXPRESS FROM HYDERABAD TO BENGALURU-----------
-- ")
t1 = train()
t1.display()
type = int(input("DO YOU WANT TO BOOK OR CANCEL TICKETS? PRESS 1 FOR
BOOKING AND 2 FOR CANCELLING:"))
tier = int(input("PRESS 1 FOR TIER-1 2 FOR TIER-2 AND 3 FOR TIER-3:"))
no_of_seats = int(input("ENTER THE NO.OF SEATS:"))
if (tier in range(1,4)) and (type in range(1,3)):
    r1 = reservation(type,tier,no_of_seats)
```

```
  r1.booking()
else:
   print("PLEASE ENTER VALID NUMBER FOR BOOKING/CANCELLING OR TIER SELECTION!!")
print("------------------------------------THANKYOU----------------------------------------")
```

OUTPUT :



```
-----------------------WELCOME TO INDIAN RAILWAYS--------------------
--------------RAJADHANI EXPRESS FROM HYDERABAD TO BENGALURU-------------

THE TOTAL NO.OF SEATS IN EACH TIER:
TIER-1(SLEEPING COACH AND AC):100                       PRICE OF EACH TICKET(FOR MEN,WOMEN AND CHILDREN):500/-
TIER-2(SLEEPING COACH NON AC):200                       PRICE OF EACH TICKET(FOR MEN,WOMEN AND CHILDREN):250/-
TIER-3(NO SLEEPING COACH AND NON AC):500         PRICE OF EACH TICKET(FOR MEN,WOMEN AND CHILDREN):100/-
DO YOU WANT TO BOOK OR CANCEL TICKETS? PRESS 1 FOR BOOKING AND 2 FOR CANCELLING:1
PRESS 1 FOR TIER-1 2 FOR TIER-2 AND 3 FOR TIER-3:2
ENTER THE NO.OF SEATS:3
----------------------------------------------------------------------------------------------------
BOOKING OF 3 SEATS IN TIER-2 IS SUCCESSFULL
TOTAL AMOUNT TO BE PAID IS:750/-
SEATING LOCATIONS ARE:
249 - E  *  250 - E  *  251 - E  *
DATE AND TIMINGS: 7-7-2022 MONDAY
ARR TIME:12:00PM
DEP TIME:1:00PM
JOURNEY DURATION:4 hrs
-----------------------------------THANKYOU----------------------------------------
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

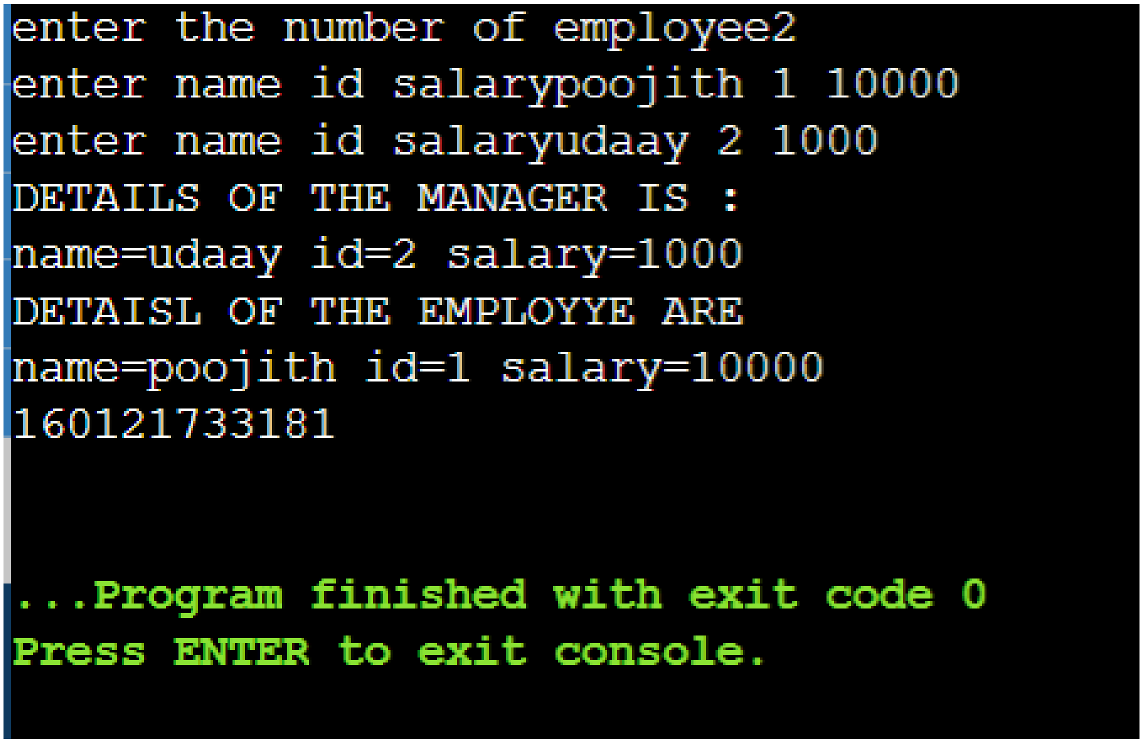
QUESTION 81:

Write a program that extends the class Employee .Derive a class manager from employee so that it lists all the details of manager as well as the detailsof the employees working under the manager.

CODE :

```
from random import *


class Employee:
    l = []

    def __init__(self, name, _id, salary):
        self.name = name
        self._id = _id
        self.salary = salary
        Employee.l.append(self)
class Manager(Employee):
    def display(self):
        h = randint(0, len(Employee.l) - 1)
        print("DETAILS OF THE MANAGER IS :")
        print(f"name={self[h].name} id={self[h]._id} salary={self[h].salary}")
        print("DETAISL OF THE EMPLOYYE ARE")
        for m in range(len(Employee.l)):
            if (h != m):
                print(f"name={self[m].name} id={self[m]._id} salary={self[m].salary}")
k = []
n = int(input("enter the number of employee"))
for i in range(n):
    x = input("enter name id salary").split()
    k.append(x)
for i in range(n):
    j = Employee(k[i][0], k[i][1], k[i][2])
Manager.display(Employee.l)
```

OUTPUT :



```
enter the number of employee2
enter name id salarypoojith 1 10000
enter name id salaryudaay 2 1000
DETAILS OF THE MANAGER IS :
name=udaay id=2 salary=1000
DETAISL OF THE EMPLOYYE ARE
name=poojith id=1 salary=10000
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

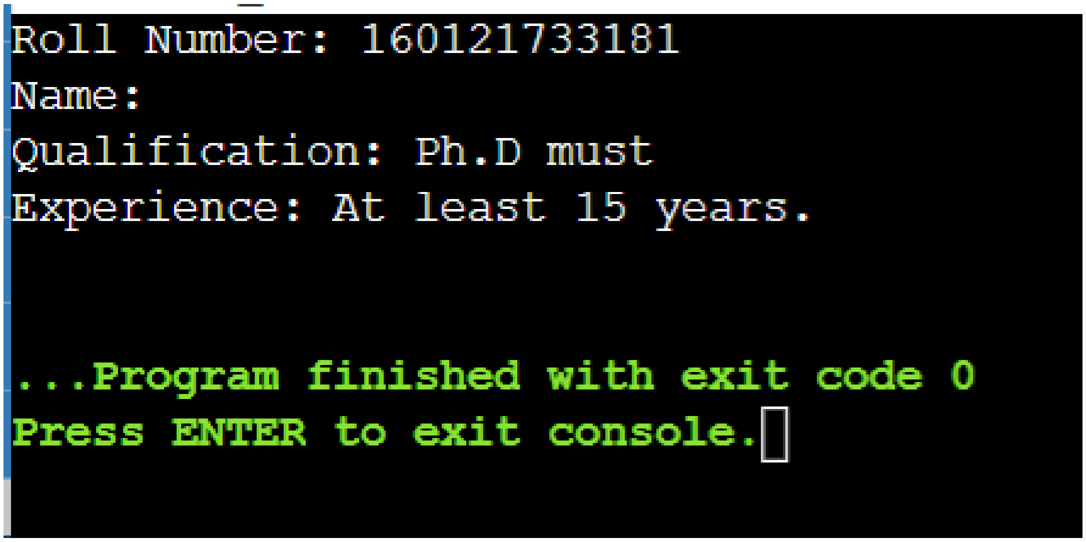
QUESTION 82 :

Write a Program to demonstrate multi level inheritance

CODE :

```
print("Roll Number: 160121733181")
class Person:
   def name(self):
       print("Name: ")
class Teacher(Person):
   def qualification(self):
       print("Qualification: Ph.D must")
class HOD(Teacher):
   def experience(self):
       print("Experience: At least 15 years.")
hod=HOD()
hod.name()
hod.qualification()
hod.experience()
```

OUTPUT :



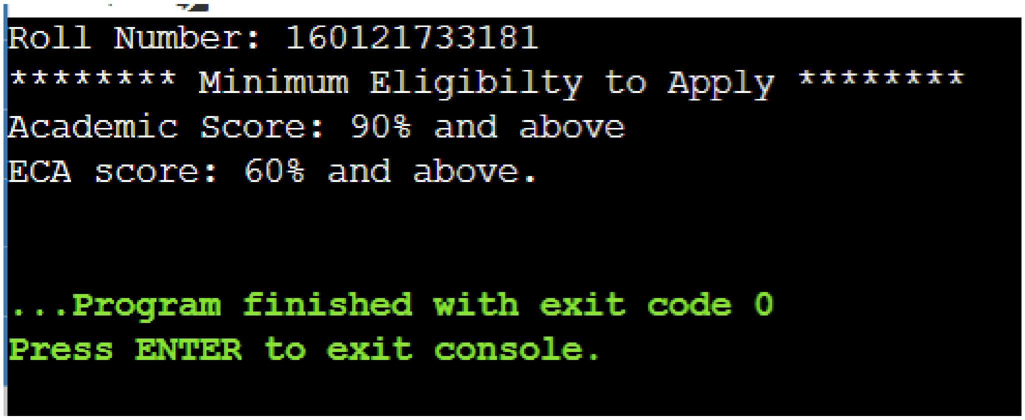
```
Roll Number: 160121733181
Name:
Qualification: Ph.D must
Experience: At least 15 years.


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 83:
Program to demonstrate multi path inheritance

CODE :

```
print("Roll Number: 160121733181")
class Student:
    def name(self):
        print("Name: ")
class Academic_Performance(Student):
    def Acad_score(self):
        print("Academic Score: 90% and above")
class ECA(Student):
    def ECA_score(self):
        print("ECA score: 60% and above.")
class Result(Academic_Performance, ECA):
    def Eligibility(self):
        print("******** Minimum Eligibilty to Apply ********")
        self.Acad_score()
        self.ECA_score()
R=Result()
R.Eligibility()
```

OUTPUT :



```
Roll Number: 160121733181
******** Minimum Eligibilty to Apply ********
Academic Score: 90% and above
ECA score: 60% and above.


...Program finished with exit code 0
Press ENTER to exit console.
```

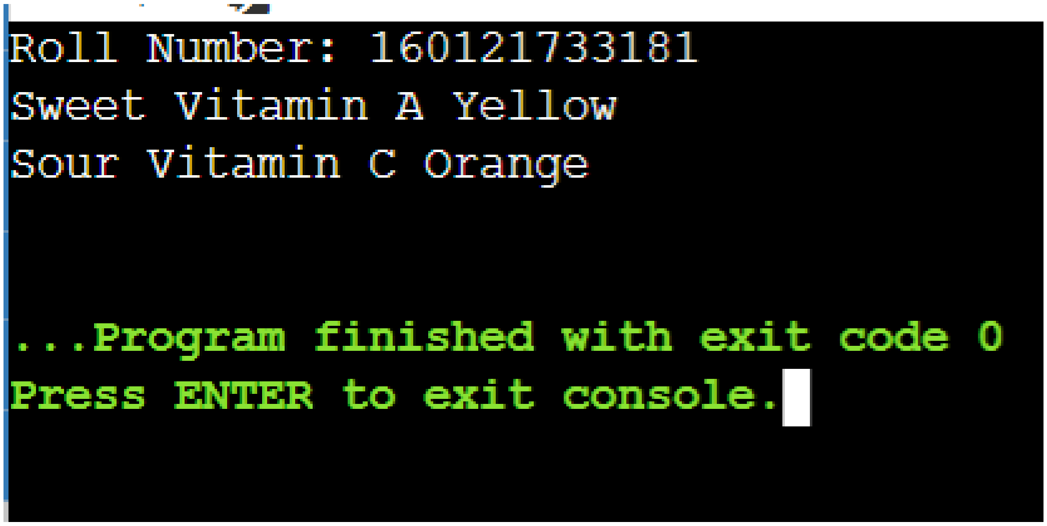
QUESTION 84 :

Program to illustrate the concept of abstract class.

CODE :

```
class Fruit:
    def taste(self):
        raise NotImplementedError()
    def rich_in(self):
        raise NotImplementedError()
    def colour(self):
        raise NotImplementedError()
class Mango(Fruit):
    def taste(self):
        return "Sweet"
    def rich_in(self):
        return "VitaminA"
    def colour(self):
        return "Yellow"
class Orange(Fruit):
    def taste(self):
         return "Sour"
    def rich_in(self):
         return "VitaminC"
    def colour(self):
         return "Orange"
Alphanso=Mango()
print(Alphanso.taste(),Alphanso.rich_in(),Alphanso.colour())
Org=Orange()
print(Org.taste(),Org.rich_in(),Org.colour())
```

OUTPUT :



```
Roll Number: 160121733181
Sweet Vitamin A Yellow
Sour Vitamin C Orange


...Program finished with exit code 0
Press ENTER to exit console.
```

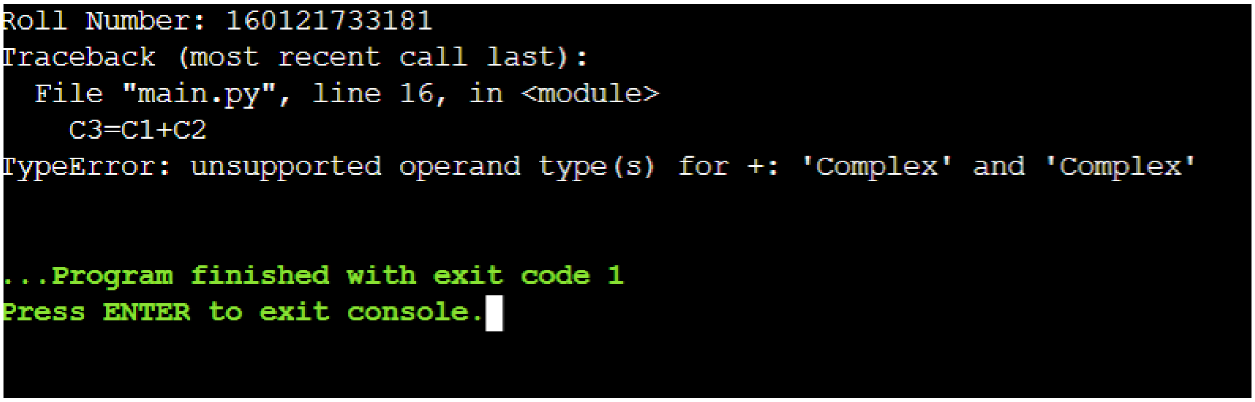
QUESTION 85 :

Program to add two complex numbers without overloading the + operator

CODE :

```
class Complex:
   def __init__(self):
      self.real=0
      self.imag=0
   def setValue(self,real,imag):
      self.real=real
      self.imag=imag
   def display(self):
      print("(",self.real,"+",self.imag,"i)")
C1=Complex()
C1.setValue(1,2)
C2=Complex()
C2.setValue(3,4)
C3=Complex()
C3=C1+C2
C3.display()
```

OUTPUT :



```
Roll Number: 160121733181
Traceback (most recent call last):
  File "main.py", line 16, in <module>
    C3=C1+C2
TypeError: unsupported operand type(s) for +: 'Complex' and 'Complex'


...Program finished with exit code 1
Press ENTER to exit console.
```

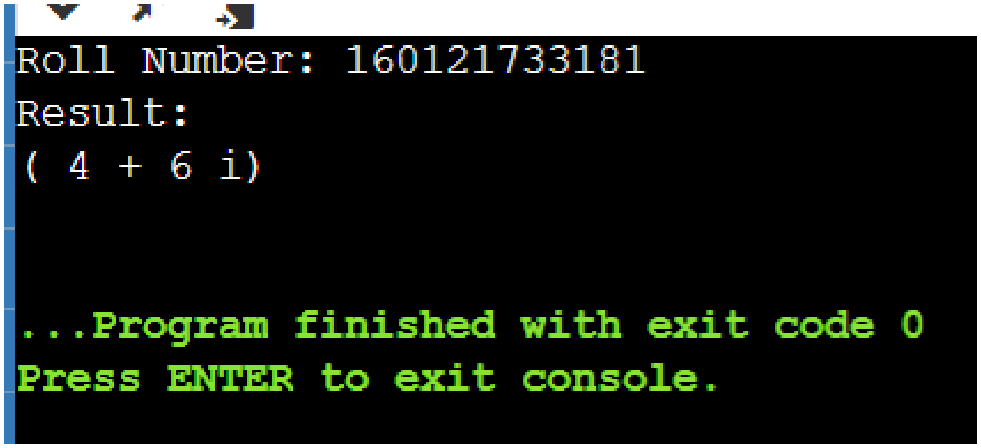
QUESTION 86 :

Program to overload the +operator on a complex object.

CODE :

```
class Complex:
   def __init__(self):
       self.real=0
       self.imag=0
   def setValue(self,real,imag):
       self.real=real
       self.imag=imag

   def __add__(self,c):
       temp=Complex()
       temp.real=self.real+c.real
       temp.imag= self.imag+c.imag
       return temp

   def display(self):
       print("(",self.real,"+",self.imag,"i)")
C1=Complex()
C1.setValue(1,2)
C2=Complex()
C2.setValue(3,4)
C3=Complex()
C3=C1+C2
print("RESULT = ")
C3.display())
```

OUTPUT :



```
Roll Number: 160121733181
Result:
( 4 + 6 i)


...Program finished with exit code 0
Press ENTER to exit console.
```

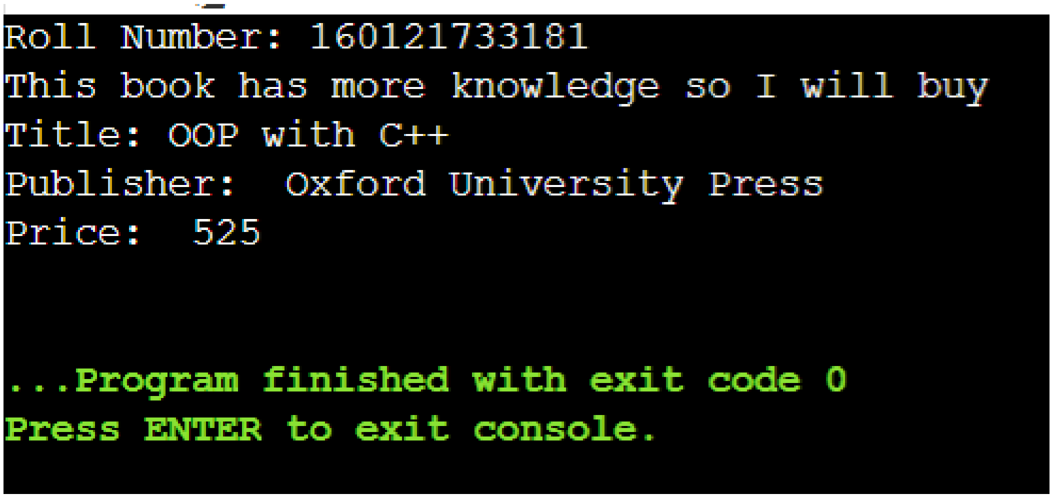
QUESTION 87 :

Program to compare two objects of user defined class type

CODE :

```
class Book:
    def __init__(self):
        title= ""
        publisher= ""
        price=0
    def set(self,title,publisher,price):
        self.title=title
        self.publisher=publisher
        self.price=price
    def display(self):
       print("TITLE:",self.title)
       print("PUBLISHER:",self.publisher)
       print("PRICE:",self.price)
    def __gt__(self,B):
        if self.price>B.price:
           return True
        else :
           return False
B1 =Book()
B1.set("OOP withC++","Oxford University Press",525)
B2=Book()
B2.set("Let usC++","BPB",300)
if B1>B2:
    print("This book has more knowledge so I will buy")
    B1.display()
```

OUTPUT :



```
Roll Number: 160121733181
This book has more knowledge so I will buy
Title: OOP with C++
Publisher:  Oxford University Press
Price:  525


...Program finished with exit code 0
Press ENTER to exit console.
```

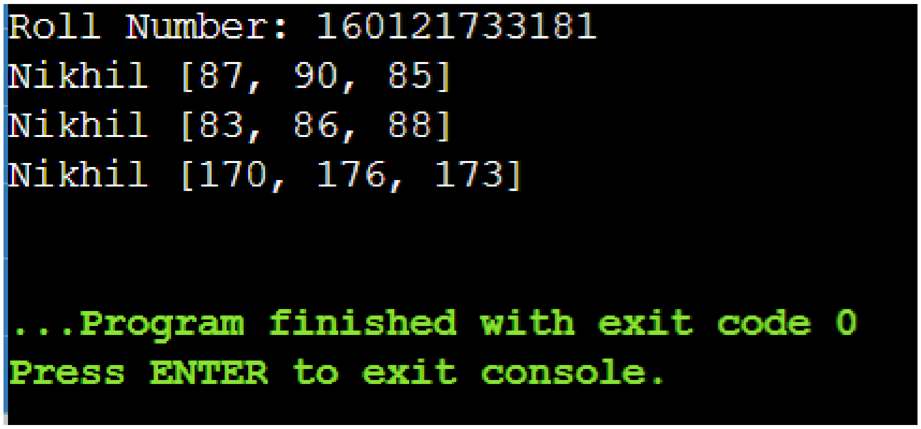
QUESTION 88 :

Program that overloads the + operator on a class student that has attributes name and marks.

CODE :

```
class Student:
   def __init__(self,name,marks):
      self.name=name
      self.marks=marks
   def display(self):
      print(self.name,self.marks)
   def __add__(self,S):
      Temp=Student(S.name,[])
      for i  in range(len(self.marks)):
         Temp.marks.append(self.marks[i]+S.marks[i])
      return Temp
S1=Student("Nikhil",[87,90,85])
S2=Student("SHIVA",[83,86,88])
S1.display()
S2.display()
S3=Student('',[])
S3=S1+S2
S3.display()
```

OUTPUT :



```
Roll Number: 160121733181
Nikhil [87, 90, 85]
Nikhil [83, 86, 88]
Nikhil [170, 176, 173]


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 89 :

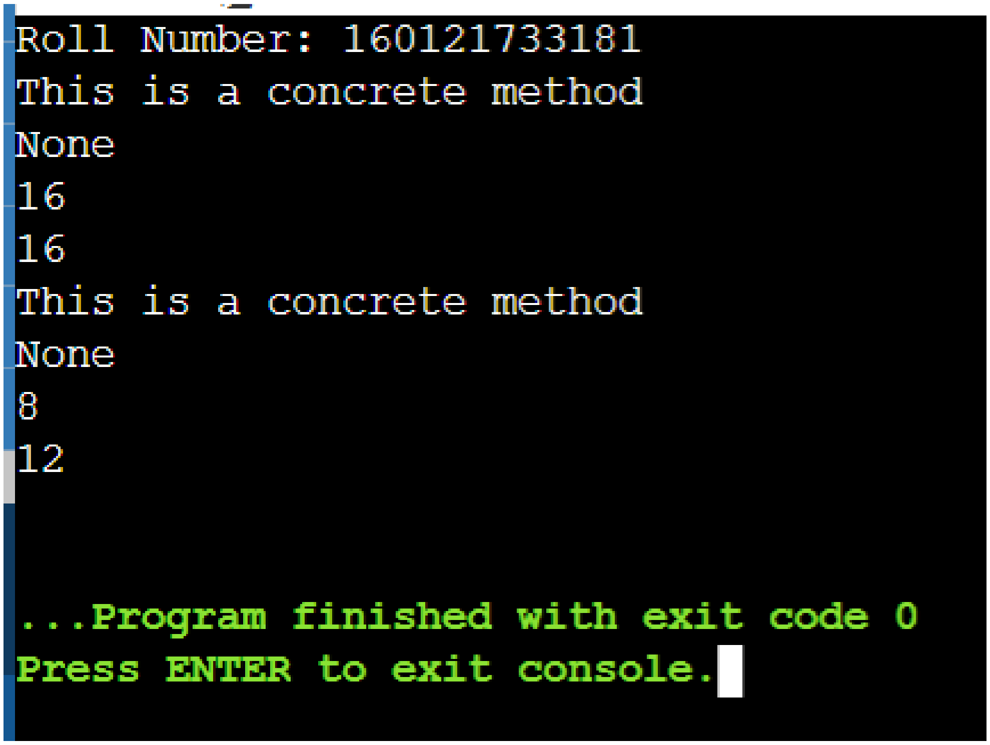
Program to demonstrate abstract class.

CODE :

```
print("Roll Number: 160121733181")
from abc import ABC,abstractmethod
class Shape(ABC):
    def common(self):
        print("This is a concrete method")
    @abstractmethod
    def area(self):
        pass
    @abstractmethod
    def perimeter(self):
        pass
class Square(Shape):
    def __init__(self,side):
        self.__side=side
    def area(self):
        return self.__side*self.__side
    def perimeter(self):
        return 4*self.__side
class Rectangle(Shape):
    def __init__(self,length,breath):
        self.__length=length
        self.__breath=breath
    def area(self):
        return self.__length*self.__breath
    def perimeter(self):
        return 2*(self.__length+self.__breath)
S1=Square(4)
print(S1.common())
print(S1.area())
print(S1.perimeter())
R1=Rectangle(2,4)
print(R1.common())
print(R1.area())
```

print(R1.perimeter())

OUTPUT :



```
Roll Number: 160121733181
This is a concrete method
None
16
16
This is a concrete method
None
8
12


...Program finished with exit code 0
Press ENTER to exit console.
```

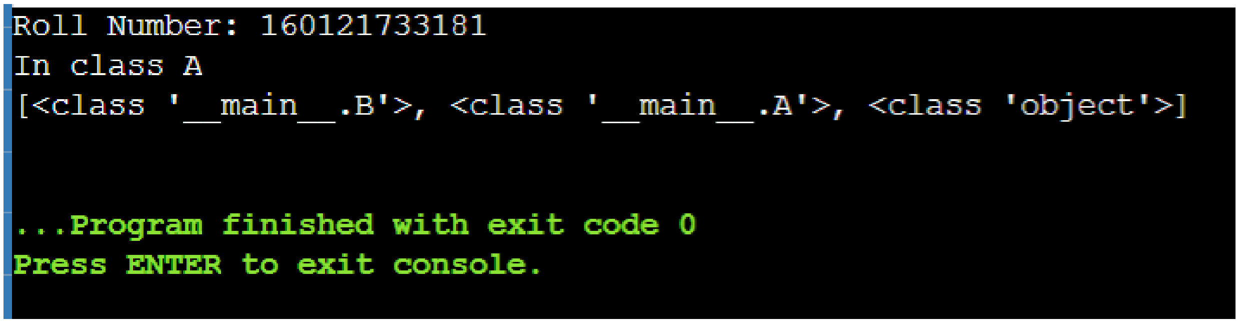
QUESTION 90 :

Program to demonstrate MRO of single inheritance

CODE :

```
class A:
    def display(self):
        print('In ClassA')
class B(A):
    def show(self):
        print('In ClassB')
x=B()
x.display()
#Display MRO of ClassB
print(B.mro())
```

OUTPUT :



```
Roll Number: 160121733181
In class A
[<class '__main__.B'>, <class '__main__.A'>, <class 'object'>]


...Program finished with exit code 0
Press ENTER to exit console.
```

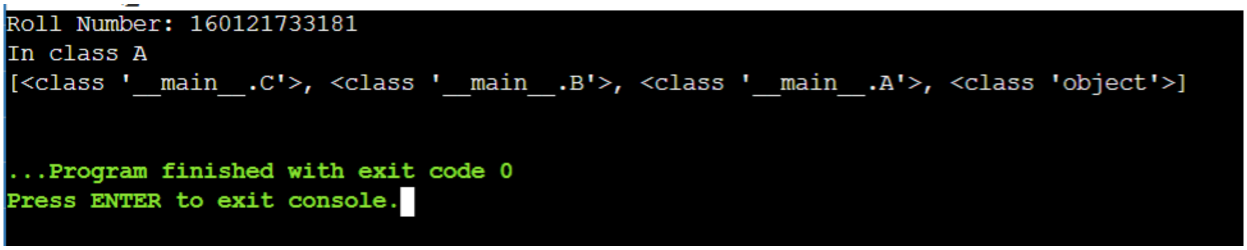
QUESTION 91 :

Program to demonstrate MRO of multilevel inheritance.

CODE :

```
class A:
   def display_x(self):
        print('In ClassA')
class B(A):
    def display_y(self):
        print('In ClassB')
class C(B):
    def display_z(self):
        print('In ClassC')
x=C()
x.display_x()
#Display MRO of ClassC
print(C.mro())
```

OUTPUT :



```
Roll Number: 160121733181
In class A
[<class '__main__.C'>, <class '__main__.B'>, <class '__main__.A'>, <class 'object'>]


...Program finished with exit code 0
Press ENTER to exit console.
```

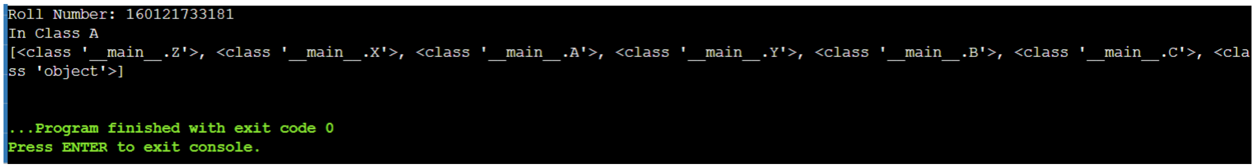
QUESTION 92 :

Program to demonstrate MRO of Multiple Inheritance

CODE :

```
class A:
  def display_a(self):
       print('In ClassA')
class B:
  def display_b(self):
       print('In ClassB')
class C:
  def display_c(self):
       print('In Class(')
class X(A,B):
  def display_x(self):
       print('In ClassX')
class Y(B,C):
  def display_y(self):
       print('In ClassY')
class Z(X,Y,C):
  def display_z(self):
       print('In ClassZ')
x=Z()
x.display_a()
#Display MRO of ClassZ
print(Z.mro())
```

OUTPUT :



```
Roll Number: 160121733181
In Class A
[<class '__main__.Z'>, <class '__main__.X'>, <class '__main__.A'>, <class '__main__.Y'>, <class '__main__.B'>, <class '__main__.C'>, <cla
ss 'object'>]


...Program finished with exit code 0
Press ENTER to exit console.
```

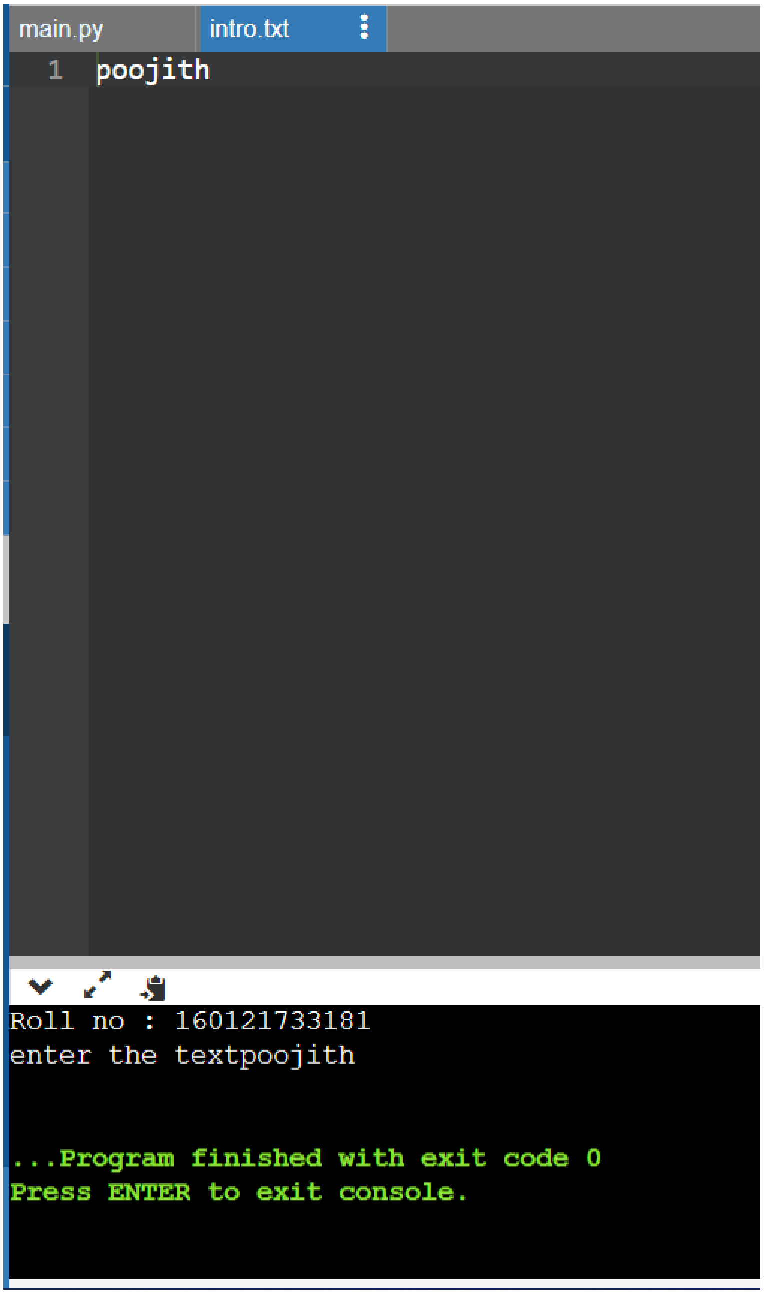
QUESTION 93 :

Create a text file "intro.txt" in python and ask the user to write a single line of text by user input.

CODE :

```
def program1() :
   f = open("intro.txt","w")
   text = input("Enter the text :")
   f.write(text)
   f.close()


program1()
```

OUTPUT :



```
poojith
```

```
Roll no : 160121733181
enter the textpoojith


...Program finished with exit code 0
Press ENTER to exit console.
```

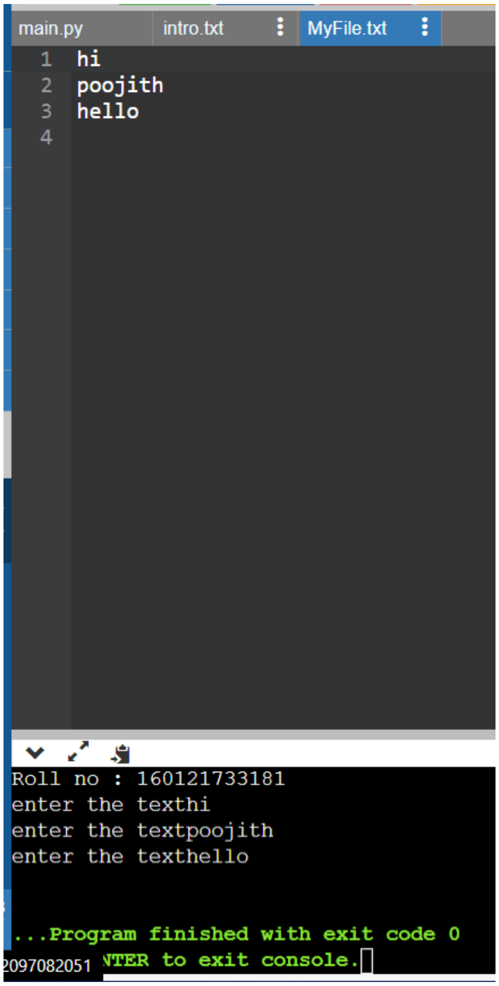
QUESTION 94 :

Create a text file “MyFile.txt” in python and ask the user to write separate 3 lines with three different input statements from the user.

CODE :

```
def program2():
   f=open("MyFile.txt","w")
   linel=input("Enter the text:")
   line2=input("Enter the text:")
   line3=input("Enter the text:")
   new_line="\n"
   f.write(linel)
   f.write(new_line)
   f.write(line2)
   f.write(new_line)
   f.write(line3)
   f.write(new_line)
   f.close()
program2()
```

OUTPUT :



```
hi
poojith
hello

```

```
Roll no : 160121733181
enter the texthi
enter the textpoojith
enter the texthello


...Program finished with exit code 0
2097082051 NTER to exit console.
```

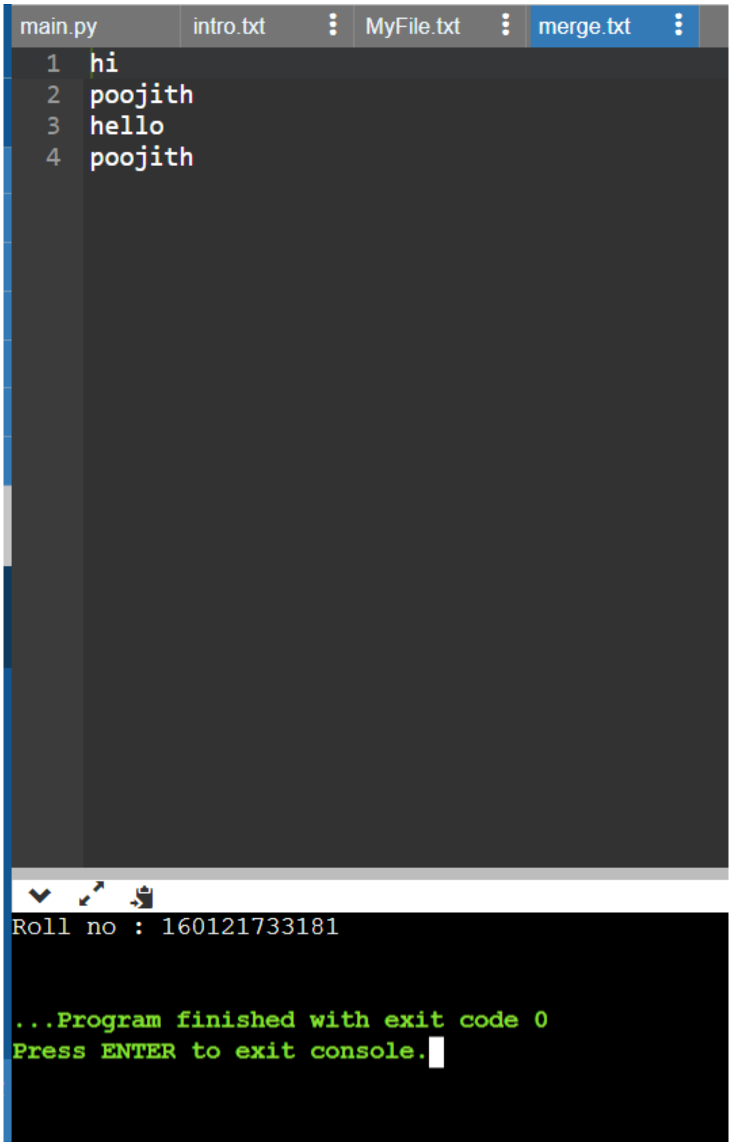
QUESTION 95 :

Write a program to read the contents of both the files created in the above programs and merge the contents into “merge.txt” .Avoid using close() function to close the files.

CODE :

```
def program3():
  with open("MyFile.txt","r")as f1:
    data=f1.read()
  with open("intro.txt","r")as f2:
    data1=f2.read()
  with open("merge.txt","w")as f3:
    f3.write(data)
    f3.write(data1)
program3()
```

OUTPUT :



```
hi
poojith
hello
poojith
```

```
Roll no : 160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

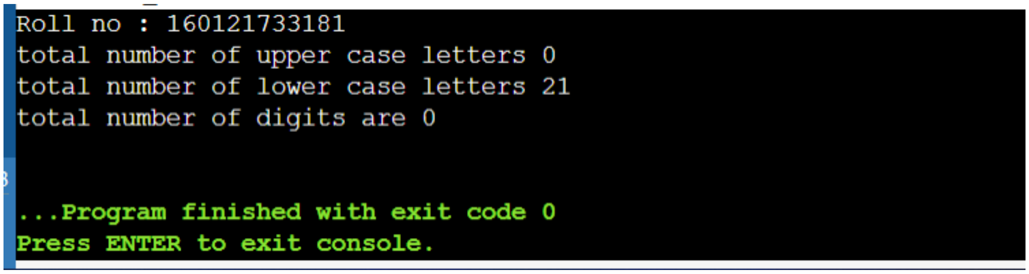
QUESTION 96 :

Count the total number of upper case , lower case ,and digits used in the text file “merge.txt”

CODE :

```
def program4():
   with open("merge.txt","r")as f1:
       data=f1.read()
       cnt_ucase=0
       cnt_lcase=0
       cnt_digits=0
       for ch in data:
           if ch.islower():
               cnt_lcase+=1
           if ch.isupper():
               cnt_ucase+=1
           if ch.isdigit() :
               cnt_digits+=1
       print("Total Number of Upper Case letters are:",cnt_ucase)
       print("Total Number of Lower Case letters are:",cnt_lcase)
       print("Total Number of digits are:",cnt_digits)
program4()
```

OUTPUT :



```
Roll no : 160121733181
total number of upper case letters 0
total number of lower case letters 21
total number of digits are 0


...Program finished with exit code 0
Press ENTER to exit console.
```

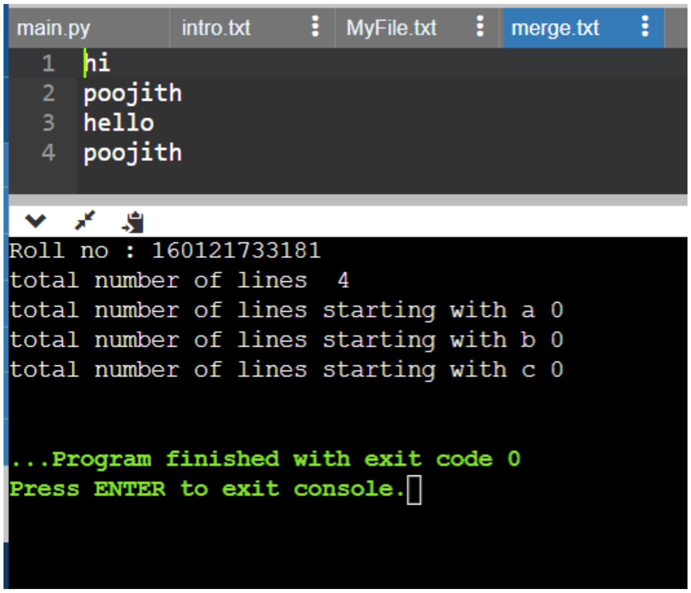
QUESTION 97 :

Wirte a program to count the total number of lines and count the total number of lines starting with ‘A’,’B’ ,’C’

CODE :

```
def programs():
  with open("merge.txt","r")as f1:
      data=f1.readlines()
      cnt_lines=0
      cnt_A=0
      cnt_B=0
      cnt_C=0
      for lines in data:
        cnt_lines+=1
        if lines[0]=='A':
          cnt_A+=1
        if lines[0]=='B':
         cnt_B+=1
        if lines[0]=='c':
          cnt_C+=1
  print("Total Number of lines are:",cnt_lines)
  print("Total Number of lines strating withAare:",cnt_A)
  print("Total Number of lines strating withBare:",cnt_B)
  print("Total Number of lines strating withCare:", cnt_C)
programs()
```

OUTPUT :



```
hi
poojith
hello
poojith
```

```
Roll no : 160121733181
total number of lines  4
total number of lines starting with a 0
total number of lines starting with b 0
total number of lines starting with c 0


...Program finished with exit code 0
Press ENTER to exit console.
```

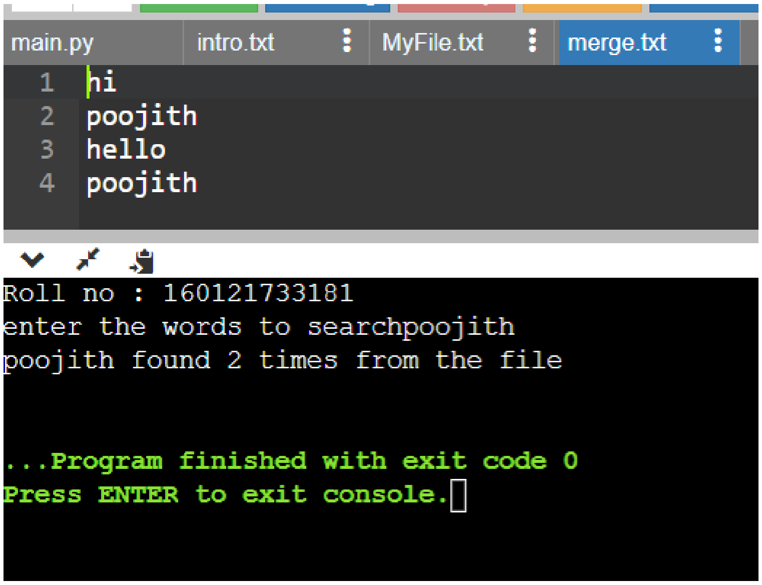
QUESTION 98 :

Find the total occurrence of a specific word from a text file

CODE :

```
def program6():
   cnt=0
   word_search=input("Enter the words to search:")
   with open("merge.txt","r")as f1:
       for data in f1:
           words=data.split()
           for word in words:
                if(word == word_search):
                    cnt+=1

   print(word_search,"found",cnt,"times from the file")
program6()
```

OUTPUT :



```
hi
poojith
hello
poojith
```

```
Roll no : 160121733181
enter the words to searchpoojith
poojith found 2 times from the file


...Program finished with exit code 0
Press ENTER to exit console.
```

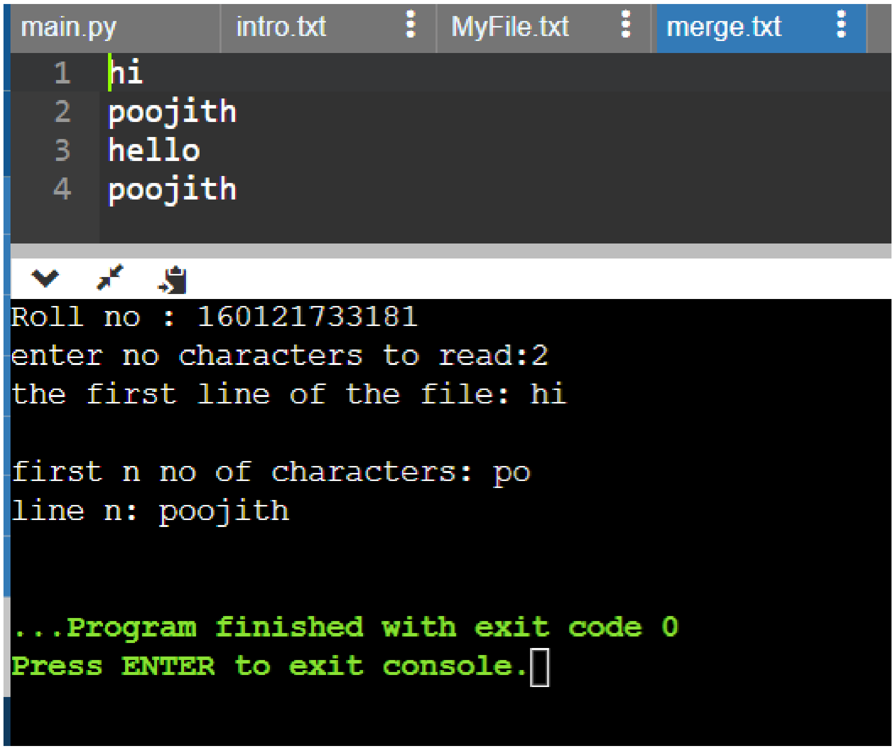
QUESTION 99 :

Read first n no letters from a text file read the first line read a specific line from a text file.

CODE :

```
def program7():
  cnt=0
  n=int(input("Enter no.characters to read:"))
  with open("merge.txt","r")as f1:
    line1=f1.readline()
    print("The first line of file:",line1)
    nchar=f1.read(n)
    print("First n no.of characters:",nchar)
    nline=f1.readlines()
    print("Line n:",nline[n])
program7()
```

OUTPUT :



```
hi
poojith
hello
poojith
```

```
Roll no : 160121733181
enter no characters to read:2
the first line of the file: hi

first n no of characters: po
line n: poojith


...Program finished with exit code 0
Press ENTER to exit console.
```

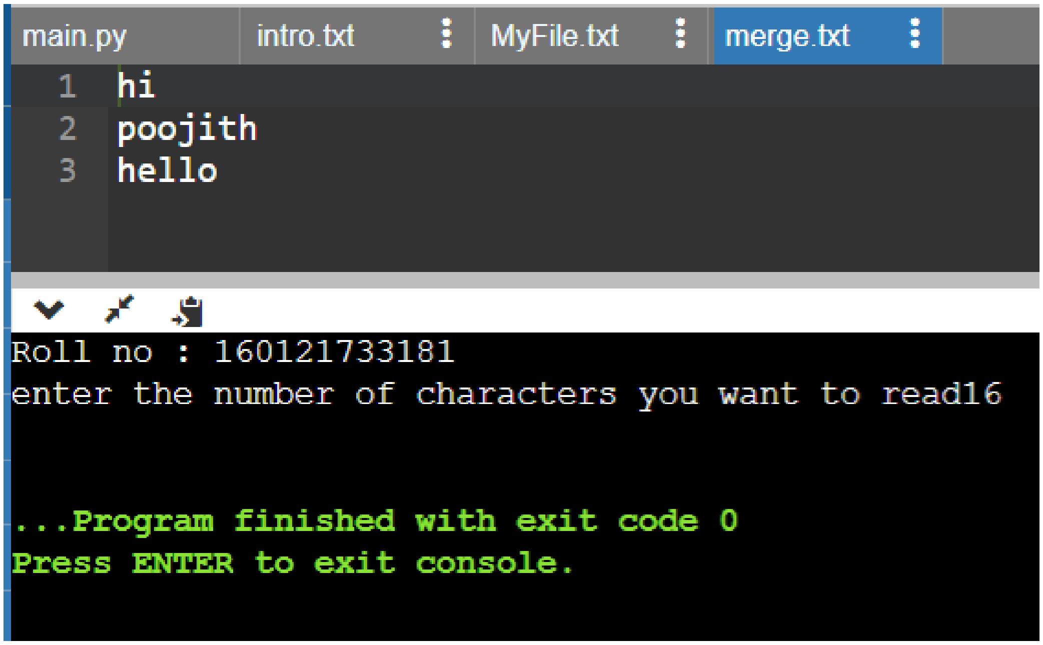
QUESTION 100 :

Replace all spaces from text with ‘,’(comma)

CODE :

```
def program8():
    cnt=0
    with open("merge.txt","r")as f1:
       data=f1.read()
       data=data.replace(",',')
    with open("merge.txt","w")as f1:
        f1.write(data)
program8()
```

OUTPUT :



```
hi
poojith
hello
```

```
Roll no : 160121733181
enter the number of characters you want to read16


...Program finished with exit code 0
Press ENTER to exit console.
```

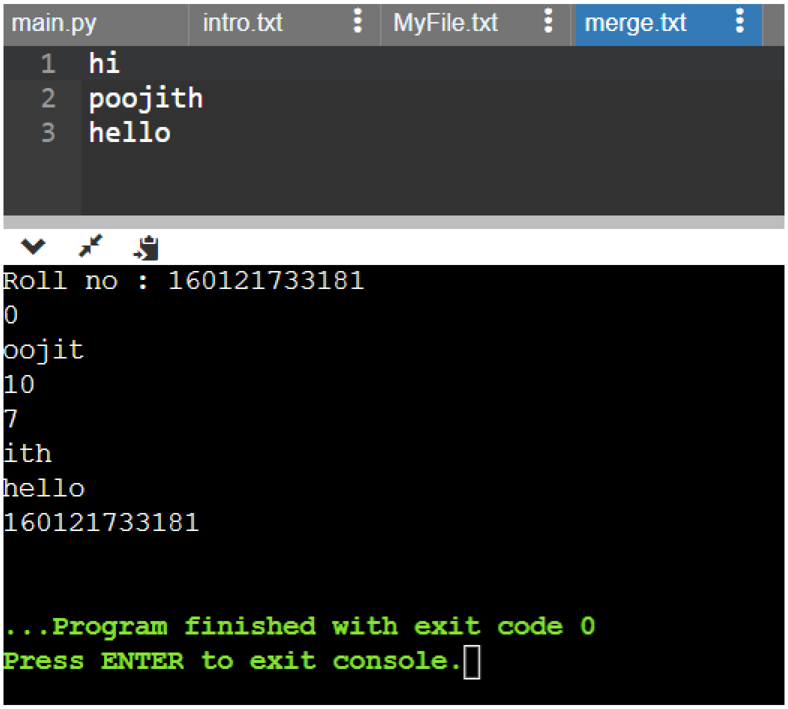
QUESTION 101 :

Write a program to know the cursor position and print the text according to below-given specifications:
1.print the intial cursor
2.move the cursor to 4th position
3.display next 5 characters
4.move the cursor to the next 10 characters
5.print the current cursor position
6.print next 10 characters form the current cursor position

CODE :

```
def program9():
  f=open("merge.txt","r")
  print(f.tell())
  f.seek(4,0)
  print(f.read(5))
  f.seek(10,0)
  print(f.tell())
  print(f.seek(7,0))
  print(f.read(10))
program9()
```

OUTPUT :



```
hi
poojith
hello
```

```
Roll no : 160121733181
0
oojit
10
7
ith
hello
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

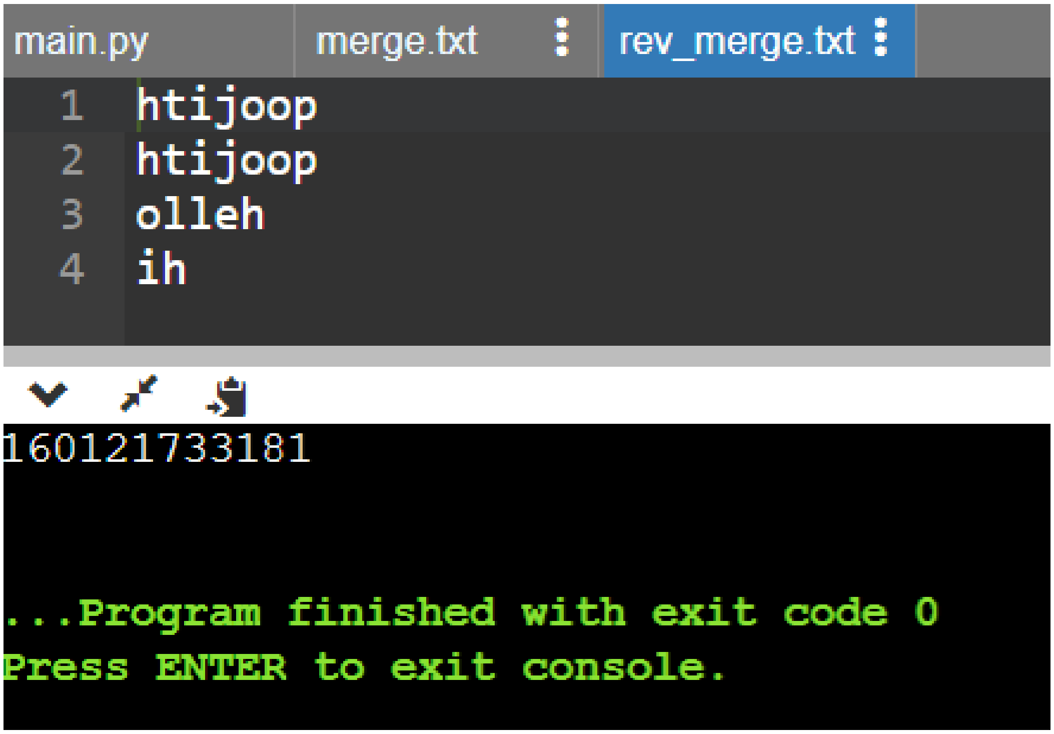
QUESTION 102 :

Write a program that reads text from a file and writes it into another file but in the reverse order.

CODE :

```
def program10():
  f=open("merge.txt","r")
  data = f.read()
  with open("rev_merge.txt","w") as f1 :
     f1.write(data[::-1])
  f.close()

program10()
```

OUTPUT :



```
htijoop
htijoop
olleh
ih
```

```
160121733181


...Program finished with exit code 0
Press ENTER to exit console.
```

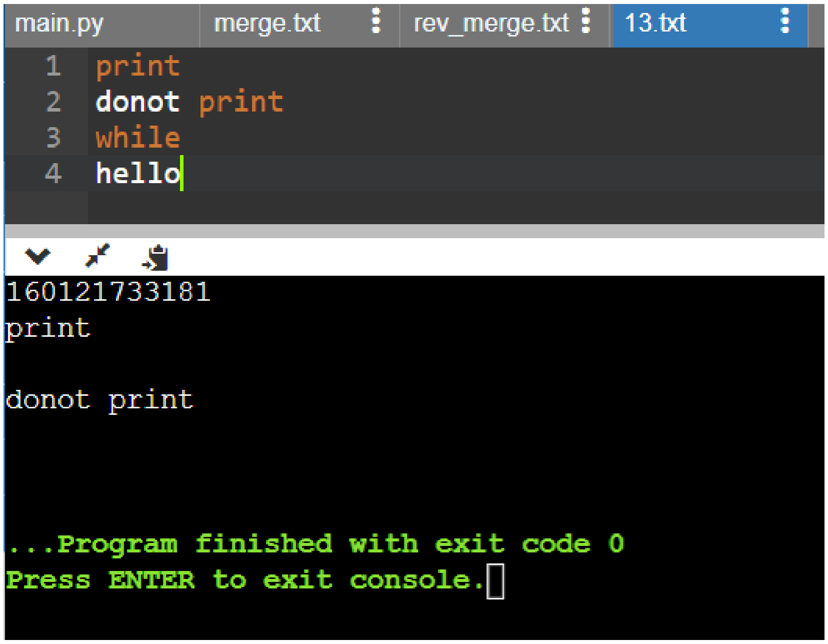
QUESTION 103 :

Write a program that reads a file and prints only those lines that has the word ‘print’.

CODE :

```
print('160121733181')
with open("13.txt") as f :
    lines = f.readlines()
    for l in lines :
        if 'print' in l :

            print(l)
```

OUTPUT :



```
print
donot print
while
hello
```

```
160121733181
print

donot print



...Program finished with exit code 0
Press ENTER to exit console.
```

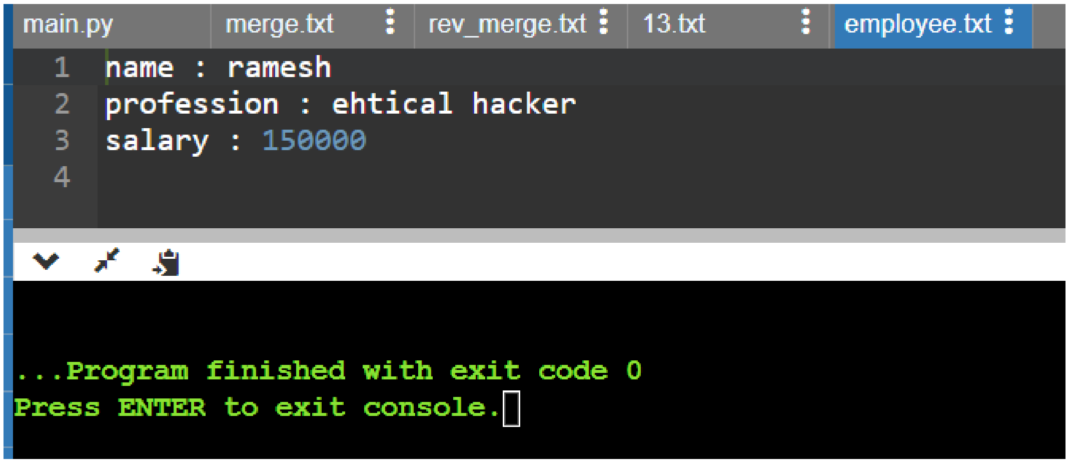
QUESTION 104 :

Write a program to edit a record stored in ‘employee’.txt file.

CODE :

```
with open("employee.txt","r") as f :
    data = f.read()
    data =data.replace("suresh","ramesh")
    data =data.replace("developer","ehtical hacker")
    data =data.replace("250000","150000")

with open("employee.txt","w") as f :
    f.write(data)
```

OUTPUT :



```
name : ramesh
profession : ehtical hacker
salary : 150000
```

```
...Program finished with exit code 0
Press ENTER to exit console.
```

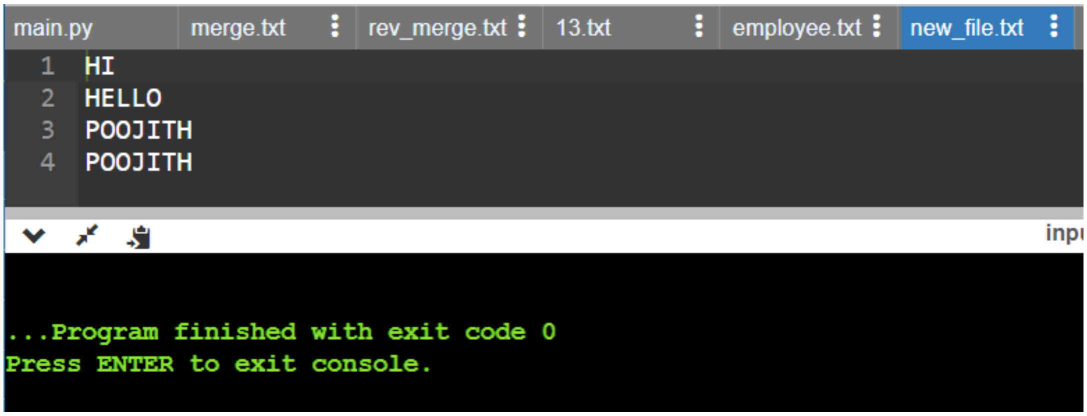
QUESTION 105 :

Write a program to read a file that contains small case characters then write these characters into another file with all lowercase characters converted into uppercase

CODE :

```
with open("merge.txt","r")as f1:
    data= f1.read()
    data= data.upper()

with open("new_file.txt","w") as  f :
    f.write(data)
```

OUTPUT :



```
HI
HELLO
POOJITH
POOJITH
```

```
...Program finished with exit code 0
Press ENTER to exit console.
```

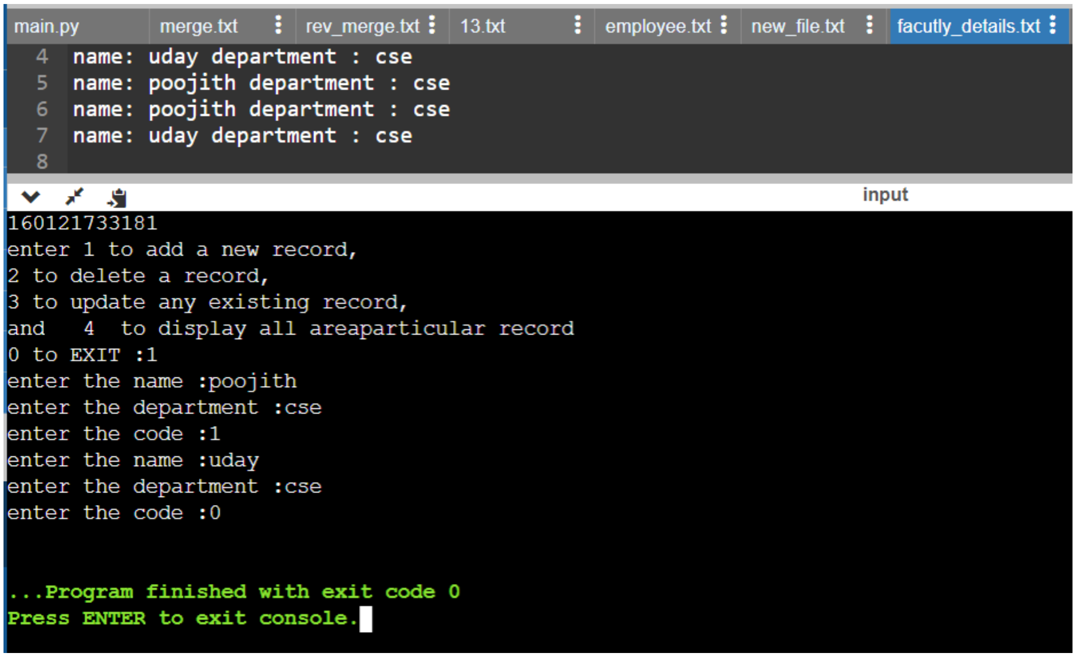
QUESTION 106 :

Write a menu driven program that reads details of a faculty. Provide options to add a new record, delete a record, update any existing record, and display all are a particular record.

CODE :

```
def record(n) :
    if n ==1 :
        name= input("enter the name :")
        dept = input("enter the department :")
        with open("facutly_details.txt","a") as f :
            f.write(f"name: {name} department : {dept} \n")
    elif n==2 :
        name = input("enter name to be deleted :")
        with open("facutly_details.txt", "r") as f:
            data = f.readlines()
            for i in data :
                if name in i :
                    data.remove(i)
        with open("facutly_details.txt", "w") as f:
             f.writelines(data)


    elif n==3 :
        name = input("enter name to be update :")
        name1 = input("enter the updated name :")
        dept = input("enter the updated department :")
        with open("facutly_details.txt", "r") as f:
            data = f.readlines()
            for i in data :
                if name in i :
                    data.remove(i)
                    data.append(f"name: {name1},department : {dept} \n")
        with open("facutly_details.txt", "w") as f:
            f.writelines(data)
    elif n==4 :
        with open("facutly_details.txt", "r") as f:
            print(f.read())



n = int(input('''enter 1 to add a new record,
2 to delete a record,
3 to update any existing record,
and   4  to display all areaparticular record
0 to EXIT :'''))
while n!=0 :
    record(n)
    n= int(input("enter the code :"))
```

OUTPUT :



```
name: uday department : cse
name: poojith department : cse
name: poojith department : cse
name: uday department : cse

```

```
160121733181
enter 1 to add a new record,
2 to delete a record,
3 to update any existing record,
and   4  to display all areaparticular record
0 to EXIT :1
enter the name :poojith
enter the department :cse
enter the code :1
enter the name :uday
enter the department :cse
enter the code :0


...Program finished with exit code 0
Press ENTER to exit console.
```

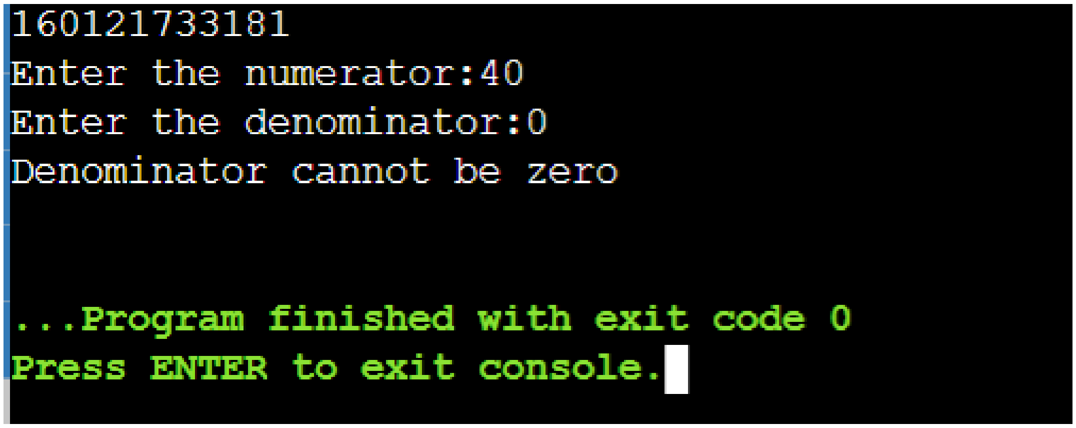
QUESTION 107 :

Program to handle the divide by zero exception.

CODE :

```
num=int(input("Enter the numerator:"))
deno=int(input("Enter the denominator:"))
try:
  quo=num/deno
  print("QUOTIENT:",quo)
except ZeroDivisionError:
  print("Denominator cannot be zero")
```

OUTPUT :



```
160121733181
Enter the numerator:40
Enter the denominator:0
Denominator cannot be zero


...Program finished with exit code 0
Press ENTER to exit console.
```

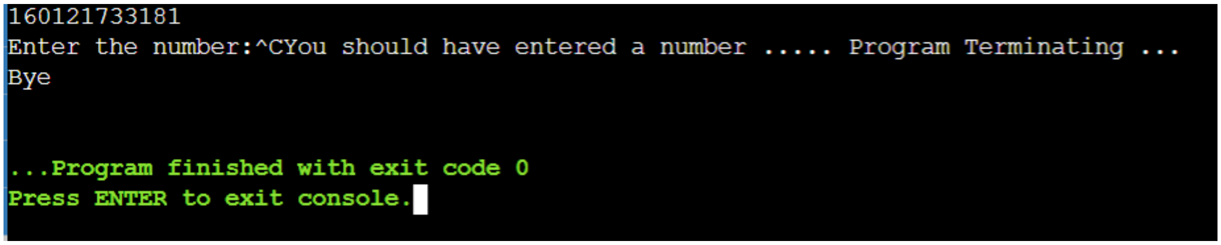
QUESTION 108 :

Program with multiple except blocks**.**

CODE :

```
try:
   num=int(input("Enter the number:"))
   print(num**2)
except(KeyboardInterrupt):
   print("You should have entered a number ..... Program Terminating ...")
except(ValueError):
   print("Please check before you enter ..... Program Terminating ...")
print("Bye")
```

OUTPUT :



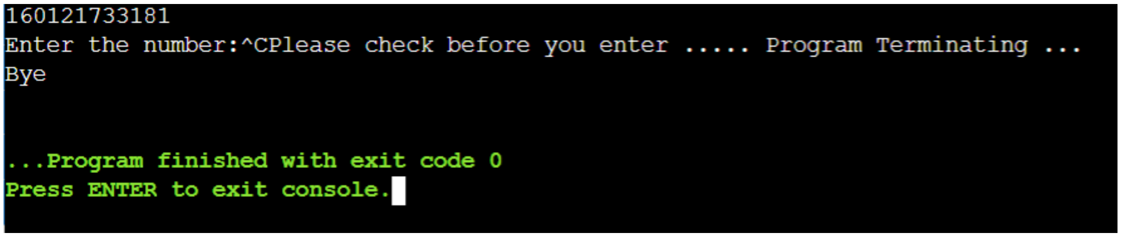
```
160121733181
Enter the number:^CYou should have entered a number ..... Program Terminating ...
Bye


...Program finished with exit code 0
Press ENTER to exit console.
```

QUESTION 109 :
Program having an exception clause handling multiple exceptions simultaneously

CODE :

```
try:
 num=int(input("Enter the number:"))
 print(num **2)
except(KeyboardInterrupt,ValueError,TypeError):
 print("Please check before you enter ..... Program Terminating ...")
print("Bye")
```

OUTPUT :



```
160121733181
Enter the number:^CPlease check before you enter ..... Program Terminating ...
Bye


...Program finished with exit code 0
Press ENTER to exit console.
```

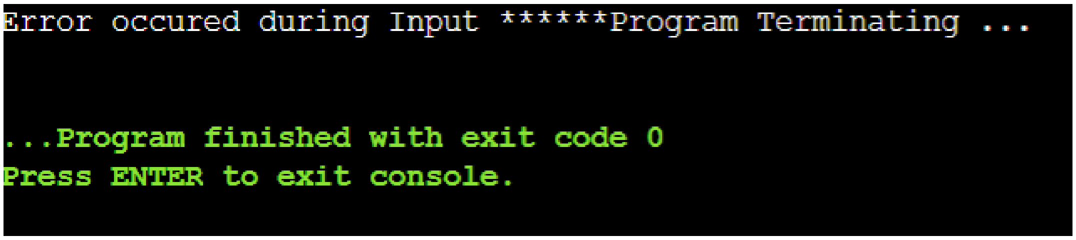
QUESTION 110 :

Program to demonstrate the use of except: block

CODE :

```
try:
  file=open('Filel.txt')
  str=file.readline()
  print(str)
except IOError:
  print("Error occured during Input ******Program Terminating ...")
except ValueError:
  print("Could not convert data to an integer.")
except:
  print("Unexpected error .... Program Terminating ...")
```

OUTPUT :



```
Error occured during Input ******Program Terminating ...


...Program finished with exit code 0
Press ENTER to exit console.
```

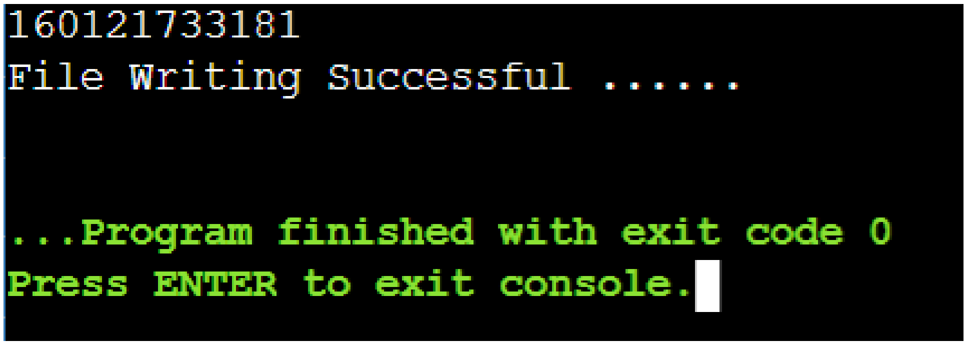
QUESTION 111 :

Write a program that opens a file and writes data to it . Handle exceptions that can be generated during the I/O operations.

CODE :

```
print(‘160121733181’)
try:
  with open('myFile.txt','w')as file:
     file.write("Hello,Good Morning !!!")
except IOError:
  print("Error working with file")
else:
    print("File Writing Successful ......")
```

OUTPUT :



```
160121733181
File Writing Successful ......


...Program finished with exit code 0
Press ENTER to exit console.
```