**Experiment No.: 1 Date : 28-10-2022**

**Aim:** To find the area of a rectangle

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

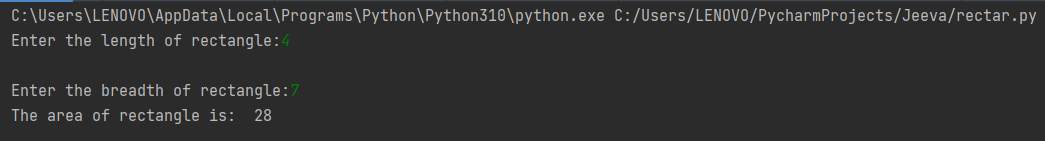
**Procedure:**

ln=int(input("Enter the length of rectangle:"))

br=int(input("\nEnter the breadth of rectangle:"))

print("The area of rectangle is: ",ln\*br)

**Output Screenshot**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 2 Date: 28-10-2022**

**Aim:** To find the area and perimeter of a circle.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

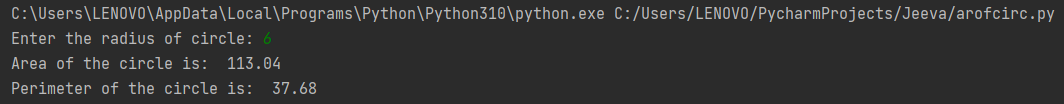
**Procedure:**

rd=int(input("Enter the radius of circle: ")) #variable rd to store radius value from user.

print("Area of the circle is: ",3.14\*(rd\*rd)) #print function to show the area of the circle using user input

print("Perimeter of the circle is: ",2\*3.14\*rd) #print function to show the area of the circle using user input

**Output Screenshot**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 3 Date: 28-10-2022**

**Aim:** To convert temperature from degree Celsius to Fahrenheit.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

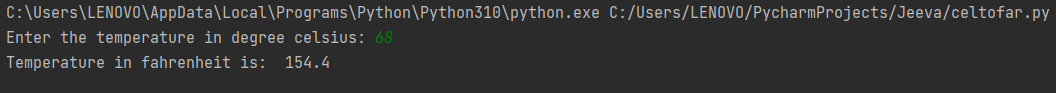
**Procedure:**

cel=int(input("Enter the temperature in degree celsius: "))

far= 9/5 \* cel + 32

print("Temperature in fahrenheit is: ",far)

**Output Screenshot**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 4 Date: 28-10-2022**

**Aim:** To covert distance from km to m.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

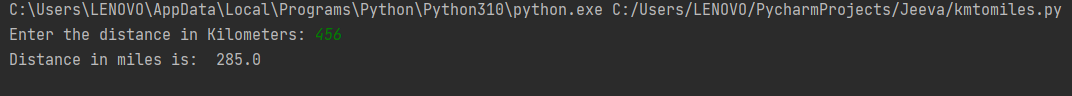
**Procedure:**

km=int(input("Enter the distance in Kilometres: "))

m=km/1.6

print("Distance in miles is: ",m)

**Output Screenshot**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 5 Date: 28-10-2022**

**Aim:** To swap 2 variables.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

**Procedure:**

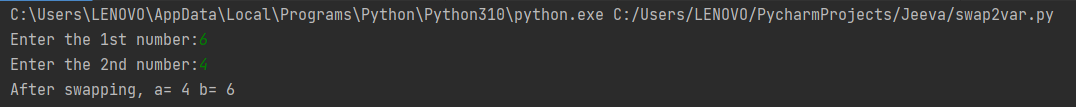
a= int(input("Enter the 1st number:"))

b= int(input("Enter the 2nd number:"))

a,b=b,a

print("After swapping, a=",a,"b=",b)

**Output Screenshot**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 6 Date: 28-10-2022**

**Aim:** To find the total percentage of a student by entering name and marks of 5 subjects.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

**Procedure:**

name =input("Enter the name of the student: ")

m1 = int(input("Total marks in Python: "))

m2 = int(input("Total marks in C: "))

m3 = int(input("Total marks in Java: "))

m4 = int(input("Total marks in C++: "))

m5 = int(input("Total marks in HTML: "))

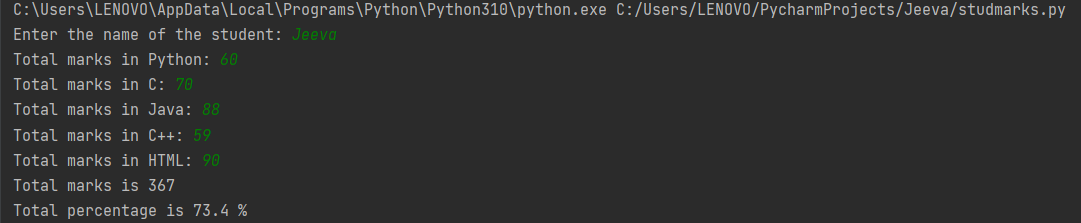
total = m1+m2+m3+m4+m5

per = total/500 \* 100

print("Total marks is",total)

print("Total percentage is",per,"%")

**Output Screenshot**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 7 Date: 28-10-2022**

**Aim:** To convert distance in feet to inches.

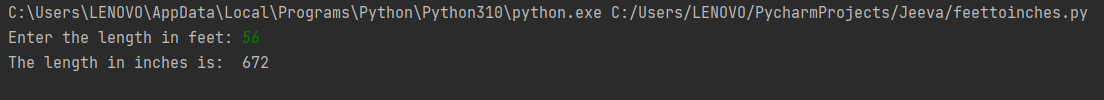
**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

**Procedure:**

ft=int(input("Enter the length in feet: "))

print("The length in inches is: ", 12\*ft)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 8 Date: 28-10-2022**

**Aim:** To calculate volume of cylinder by entering radius and height.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

**Procedure:**

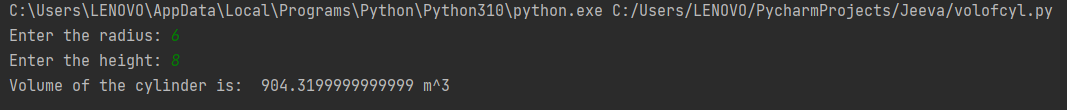
rd = int(input("Enter the radius: "))

ht = int(input("Enter the height: "))

vol = 3.14 \* rd\*rd \* ht

print("Volume of the cylinder is: ",vol,"m^3")

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 9 Date: 28-10-2022**

**Aim:** To generate a simple calculator.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

**Procedure:**

c=int(input("1.Addition\n2.Subtraction\n3.Multiplication\n4.Division\nEnter your choice:"))

a = int(input("Enter the 1st number: "))

b = int(input("Enter the 2nd number:"))

if c==1:

print("Sum = ",a+b)

elif c==2:

print("Difference = ",a-b)

elif c==3:

print("Product = ",a\*b)

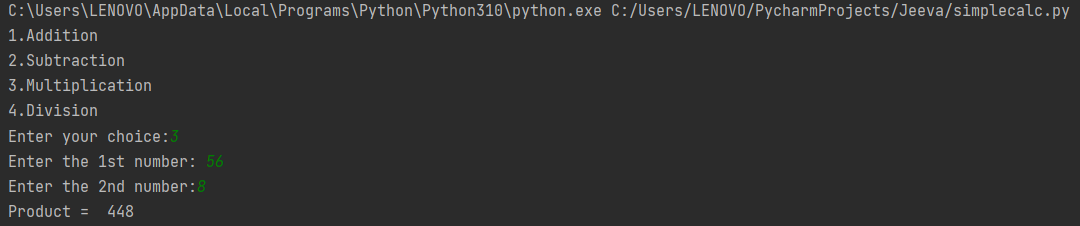
elif c==4:

print("Dividend = ",a/b)

else:

print("Invalid choice.")

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.:10 Date: 28-10-2022**

**Aim:** To calculate volume of cone by entering radius and height.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

**Procedure:**

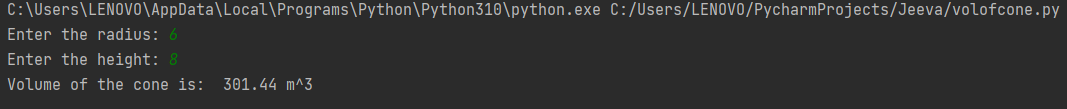
rd = int(input("Enter the radius: "))

ht = int(input("Enter the height: "))

vol = (3.14 \* rd\*rd \* ht)/3

print("Volume of the cone is: ",vol,"m^3")

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 11 Date: 31-10-2022**

**Aim:** Create string from given string by exchanging 1st and last character.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

**Procedure:**

a="PYCHARM"

print("The given string is",a)

s=a[0]

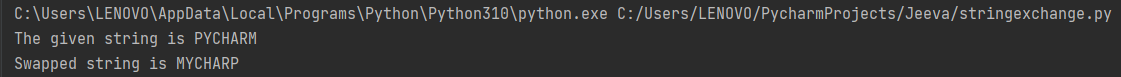
e=a[-1]

d=a[1:-1]

x=e+d+s

print("Swapped string is",x)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 12 Date: 31-10-2022**

**Aim:** To find biggest of 3 numbers using max function.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

**Procedure:**

print("Enter the 3 numbers:")

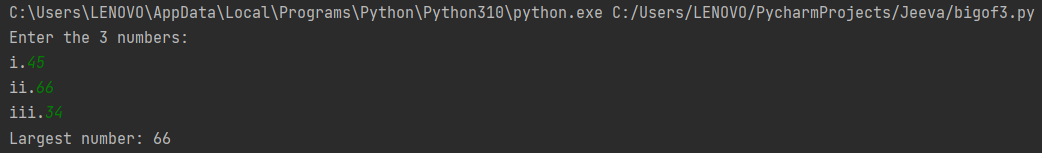
a=int(input("i."))

b=int(input("ii."))

c=int(input("iii."))

print("Largest number:",max(a,b,c))

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 13 Date: 31-10-2022**

**Aim:** To take file name from the user and print the extension.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

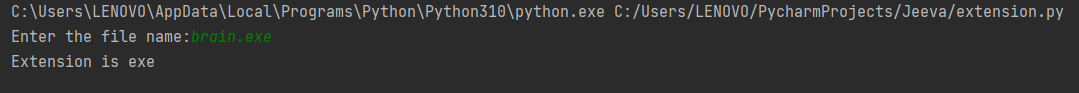
**Procedure:**

f=input("Enter the file name:")

e=f.split(".")

print("Extension is",e[-1]

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 14 Date: 31-10-2022**

**Aim:** Create list of colours where colour names are separated by ‘,’ and entered by user. Also, print 1st and last element.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

**Procedure:**

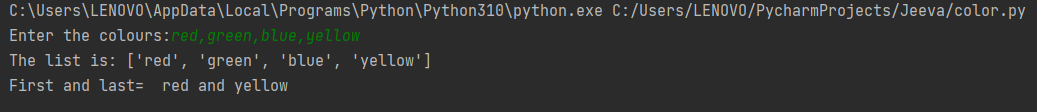
col=input("Enter the colours:")

a=col.split(",")

print("The list is:",a)

print("First and last= ",a[0],"and",a[-1])

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 15 Date: 31-10-2022**

**Aim:** Accept an integer n and print the series n+nn+nnn.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

n=int(input("Enter the integer:"))

i=1

sum=0

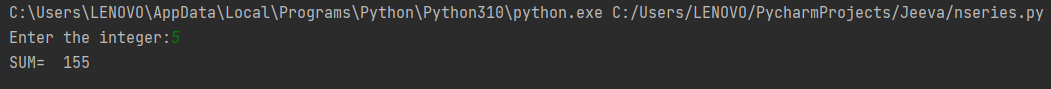
while i<=3:

sum=sum+n\*\*i

i=i+1

print("SUM= ",sum)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 16 Date: 31-10-2022**

**Aim:** Check whether the number is odd or even.

**CO1:** Understands basic of python programming language including input or output function operators. Basic and collection data types.

**Procedure:**

n=int(input("Enter the number:"))

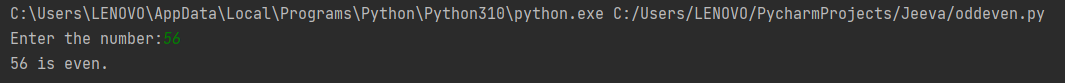
if n%2==0:

print(n,"is even.")

else:

print(n,"is odd.")

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 17 Date: 31-10-2022**

**Aim:** To check whether number is positive, negative or zero.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

n=int(input("Enter the number:"))

if n>0:

print(n,"is positive.")

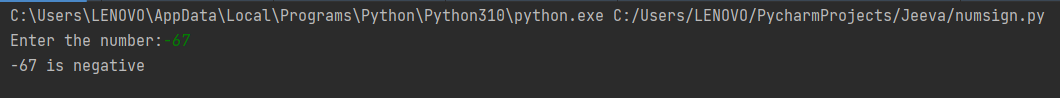
elif n==0:

print(n,"is zero")

else:

print(n,"is negative"

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 18 Date: 31-10-2022**

**Aim:** Swap characters at 1st position of 2 words separated with spaces.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

str1=input("enter the first string: ")

str2=input("enter the second string: ")

print(str1+" "+str2)

a=str1[0]

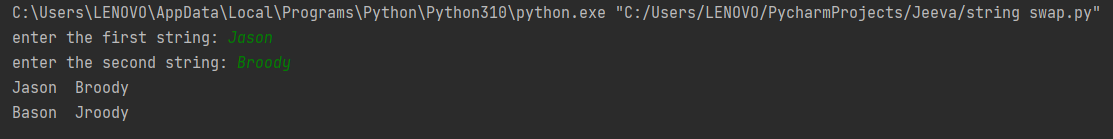
b=str2[0]

c=b+str1[1:]

d=a+str2[1:]

print(c+" "+d)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 19 Date: 31-10-2022**

**Aim:** To calculate factorial of a number.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

l=int(input("Enter the limit:"))

fact=1

i=1

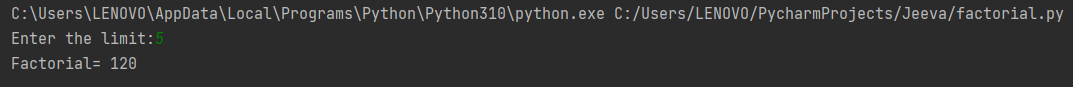
while(i<=l):

fact=fact\*i

i=i+1

print("Factorial=",fact)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 20 Date: 31-10-2022**

**Aim:** To check whether the number is palindrome of not.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

num=int(input("Enter the number"))

i=num

rev=0

while(num>0):

rem=num%10

rev=rev\*10+rem

num=num//10

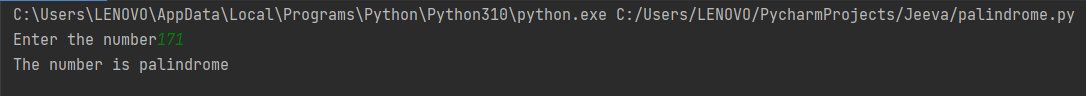
if i==rev:

print("The number is palindrome")

else:

print("The number is not palindrome")

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 21 Date: 31-10-2022**

**Aim:** To check whether the number is Armstrong or not.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

num=int(input("Enter the number:"))

i=num

sum=0

while(num>0):

rem=num%10

sum=sum+rem\*\*3

num=num//10

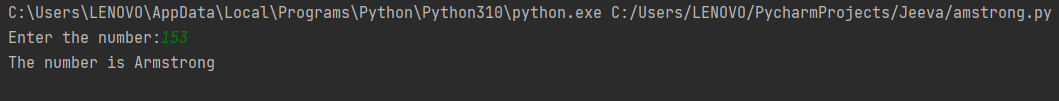
if i==sum:

print("The number is Amstrong")

else:

print("The number is not Amstrong")

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 22 Date: 31-10-2022**

**Aim:** To print fibonacii series.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

l=int(input("Enter the limit"))

a,b=0,1

i=0

print("Fiboncaii series:")

while i<l:

print(a)

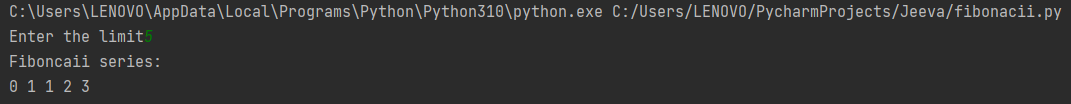
c=a+b

a=b

b=c

i=i+1

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 23 Date: 31-10-2022**

**Aim:** To print the reverse of a number.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

num = int(input("Enter the number"))

i = num

rev = 0

while num > 0:

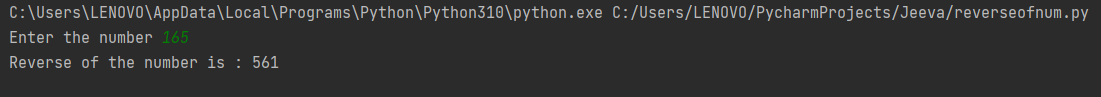
rem = num % 10

rev = rev \* 10 + rem

num = num // 10

print("Reverse of the number is :", rev

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 24 Date: 04-11-2022**

**Aim:** To print all colours from list 1 that is not included in list 2.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

c=input("Enter the colours of list 1:").split(",")

print("List 1:",c)

d=input("Enter the colours of list 2:").split(",")

print("List 1:",c)

e=set(c)

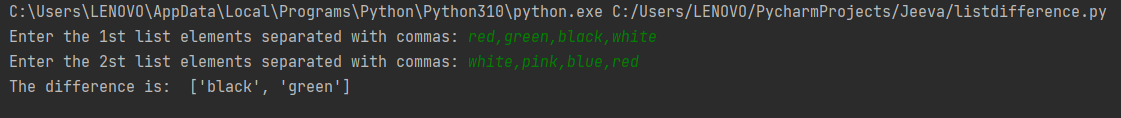
f=set(d)

sc=e.difference(f)

lc=list(sc)

print("Difference is:",lc

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 25 Date: 04-11-2022**

**Aim:** To find GCD of 2 numbers.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

a = int(input("Enter the 1st number: "))

b = int(input("Enter the 2nd number:"))

i = 1

while i <= a and i <= b:

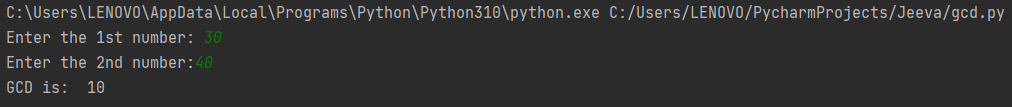
if a % i == 0 and b % i == 0:

gcd = i

i = i + 1

print("GCD is: ",gcd)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 26 Date: 04-11-2022**

**Aim:** Find sum of odd and even numbers.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

i=1

sum=0

sum2=0

n=int(input("Enter the limit"))

while(i<=n):

if i%2==0:

sum=sum+i

else:

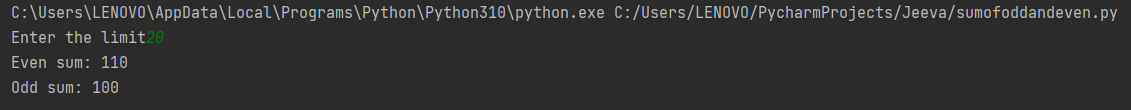
sum2=sum2+i

i=i+1

print("Even sum:",sum)

print("Odd sum:",sum2)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 27 Date: 04-11-2022**

**Aim:** To print Armstrong number within a limit.

**CO2:** Implement decision making, looping constructs and functions.

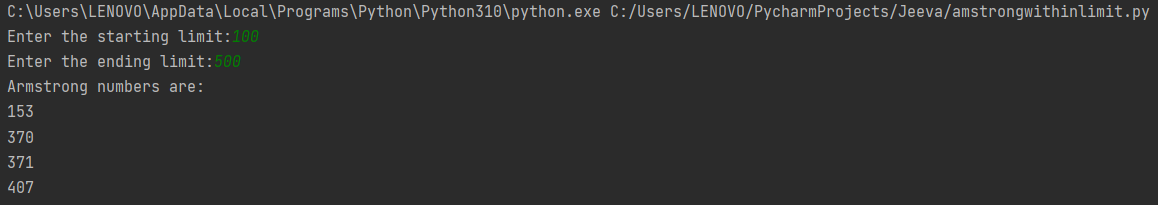
**Procedure:**

f=input("Enter the file name:")

e=f.split(".")

print("Extension is",e[-1]

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 28 Date: 04-11-2022**

**Aim:** To find sum of digits.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

n = int(input("Enter the limit:"))

i = 0

total = 0

while i <= n:

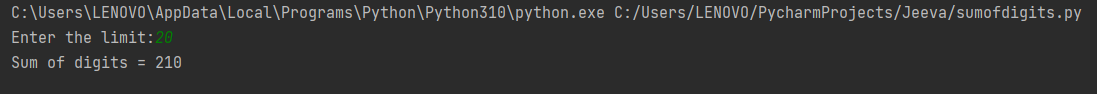
s = i

total = total + s

i = i+1

print("Sum of digits =", total)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 29 Date: 04-11-2022**

**Aim:** Construct pattern using while.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

n=int(input("Enter the limit."))

i=1

while i<=n:

j=n

while j>= i:

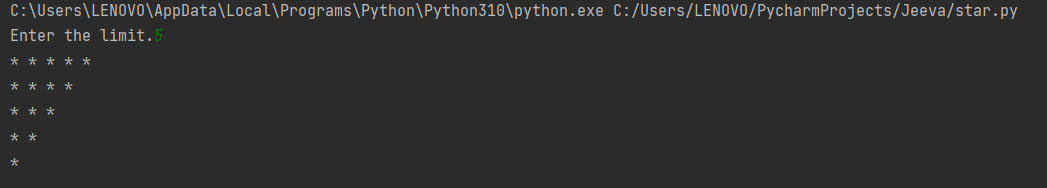
print("\*",end=" ")

j-=1

print(" ")

i+=1

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 30 Date: 04-11-2022**

**Aim:** To construct the pattern using for loop.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

n = int(input("Enter the limit:"))

for i in range (0,n):

for j in range (0,i+1):

print("\*", end=" ")

print(" ")

for i in range (0,n):

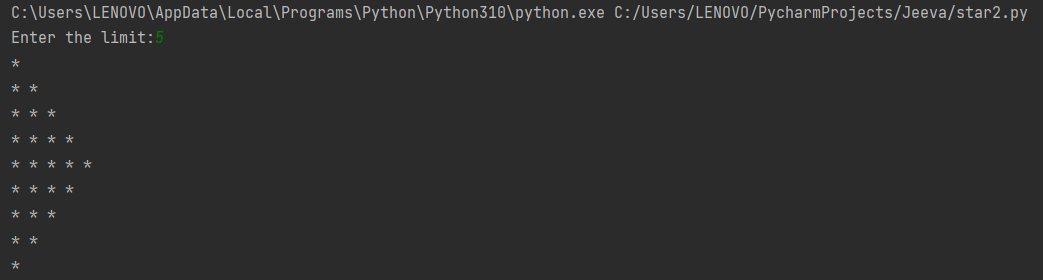
for j in range (0,j):

print("\*", end=" ")

n=n-1

print(" ")

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO1 was obtained.

**Experiment No.: 31 Date: 04-11-2022**

**Aim:** Check whether the number is prime or not.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

n=int(input("Enter the number:"))

flag=0

i=2

while i <= n/2:

if n%i == 0:

flag=1

break

i=i+1

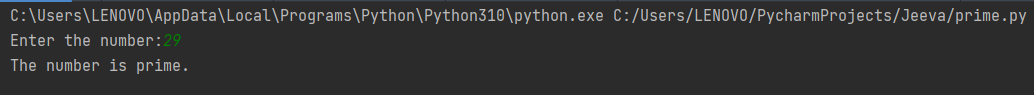
if flag == 0:

print("The number is prime.")

else:

print("The number is not prime.")

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 32 Date: 04-11-2022**

**Aim:** Display the future leap year from year to a final year entered by the user.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

n1 = 2022

n2 = int(input("Enter the limit year:"))

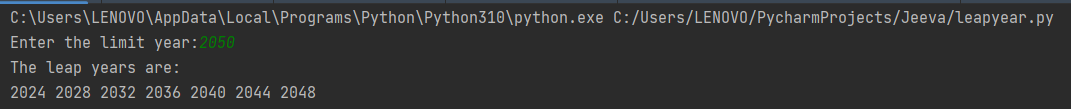
print("The leap years are:")

for i in range(n1,n2):

if i % 400 == 0 or i % 100 != 0 and i % 4 == 0:

print(i, end=" ")

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 33 Date: 04-11-2022**

**Aim:** prompt the user for a list of integers, for all values greater than 100, store “over” instead.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

l1 = []

n = int(input("Enter the limit:"))

for i in range(n):

a=int(input())

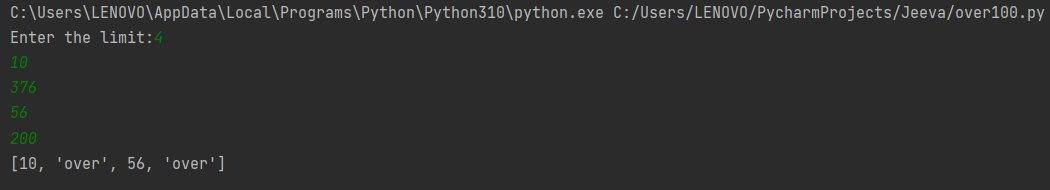
if a>100:

a="over"

l1.append(a)

print(l1)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 34 Date: 04-11-2022**

**Aim:** Display the pyramid with step number accepted from the user.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

a = int(input("Enter the number :"))

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

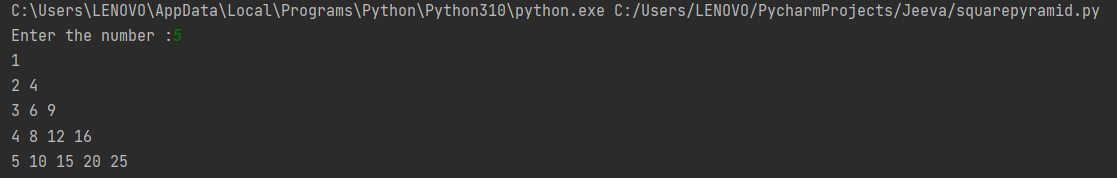
for j in range (1,i+1):

sq=i\*j

print(i\*j,end=" ")

print()

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 35 Date: 04-11-2022**

**Aim:** Form a list of vowels selected from a given word.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

w = input("Enter a string:")

l1 = []

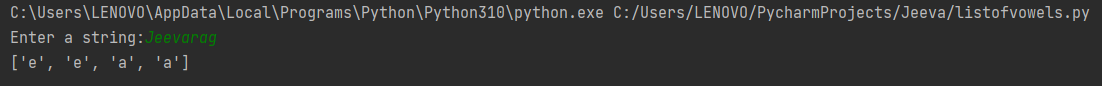
for i in w:

if i =='a' or i =='e' or i =='o' or i =='i' or i =='u':

l1.append(i)

print(l1)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 36 Date: 04-11-2022**

**Aim:** Generate an odd list of integers from a given list of integers.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

n = int(input("Enter the limit."))

li = []

i = 0

print("Enter the elements: ")

for i in range(n):

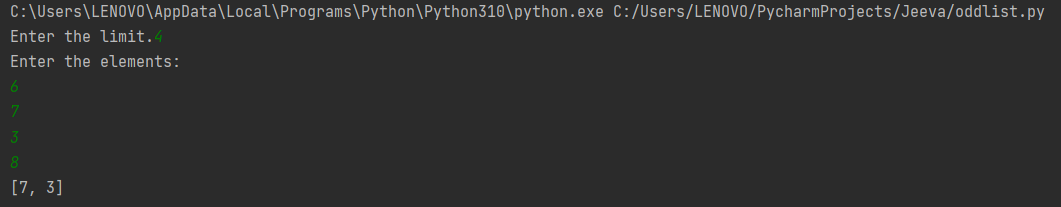
ele = int(input())

if ele % 2 != 0:

li.append(ele)

print(li)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 37 Date: 04-11-2022**

**Aim:** Square of N numbers.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

n = int(input("Enter the limit:"))

print("Squares are:")

l1 = []

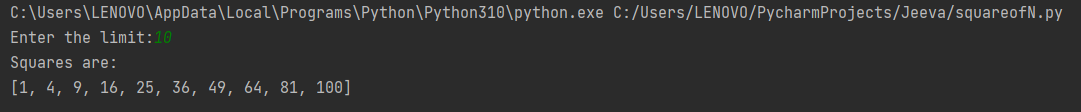
for i in range(1,n+1):

c=i\*i

l1.append(c)

print(l1

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 38 Date: 04-11-2022**

**Aim:** Generate positive list of integers.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

n = int(input("Enter the limit."))

li = []

i = 0

print("Enter the elements: ")

for i in range(n):

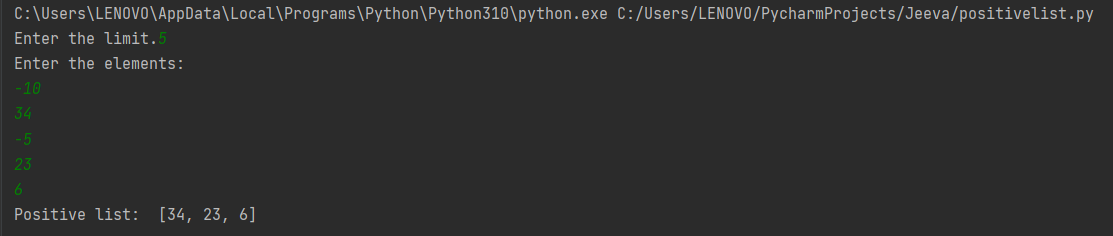
ele = int(input())

if ele > 0:

li.append(ele)

print("Positive list: ", li)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 39 Date: 04-11-2022**

**Aim:** Armstrong using for loop.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

num=int(input("Enter the number:"))

i=num

sum=0

for j in range(0,num):

rem=num%10

sum=sum+rem\*\*3

num=num//10

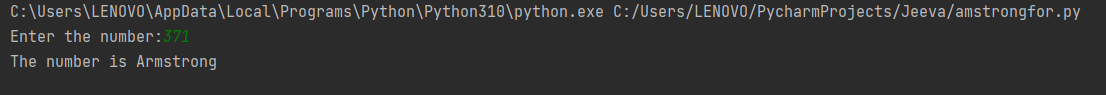
if i==sum:

print("The number is Armstrong")

else:

print("The number is not Armstrong"

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 40 Date: 04-11-2022**

**Aim:** Factorial of a number using for loop.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

l=int(input("Enter the limit:"))

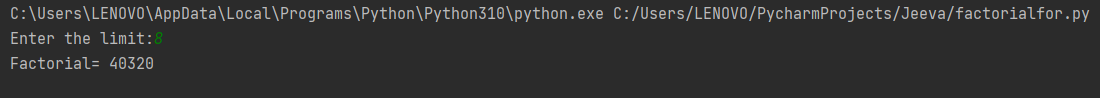
fact=1

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

fact=fact\*i

print("Factorial=",fact)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 41 Date: 04-11-2022**

**Aim:** Fibonacii series using for loop.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

l=int(input("Enter the limit"))

a,b=0,1

print("Fiboncaii series:")

for i in range(0,l):

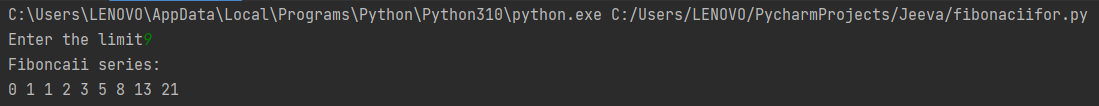
print(a, end=" ")

c=a+b

a=b

b=c

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 42 Date: 04-11-2022**

**Aim:** Generate all factors of a number.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

number = int(input("Enter a number : "))

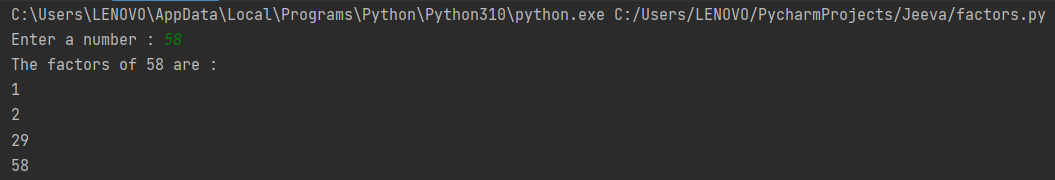
print("The factors of {} are : ".format(number))

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

if number % i == 0:

print(i)

**Output Screenshot:**



**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 43 Date: 04-11-2022**

**Aim:** Enter 2 lists of integers. Check

1. whether list is of same length
2. whether list sums to same value
3. whether any value occur in both

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

l1=int(input("Enter the size of list 1 : "))

l2=[]

for i in range (0,l1):

a=int(input("Enter Number : "))

l2.append(a)

print("\nList 1 is ;", l2)

l3=int(input("\nEnter the size of list 2 : "))

l4=[]

for i in range (0,l3):

b=int(input("Enter Number : "))

l4.append(b)

print("\nList 2 is : ", l4)

if (len(l2)==len(l4)):

print("\nThe size of both list are equal")

else:

print("\nThe size of both list are not equal")

if (sum(l2)==sum(l4)):

print("\nSum of both list is equal")

else:

print("\nSum of both list are not equal")

print("\nCommon elements in both list are : ")

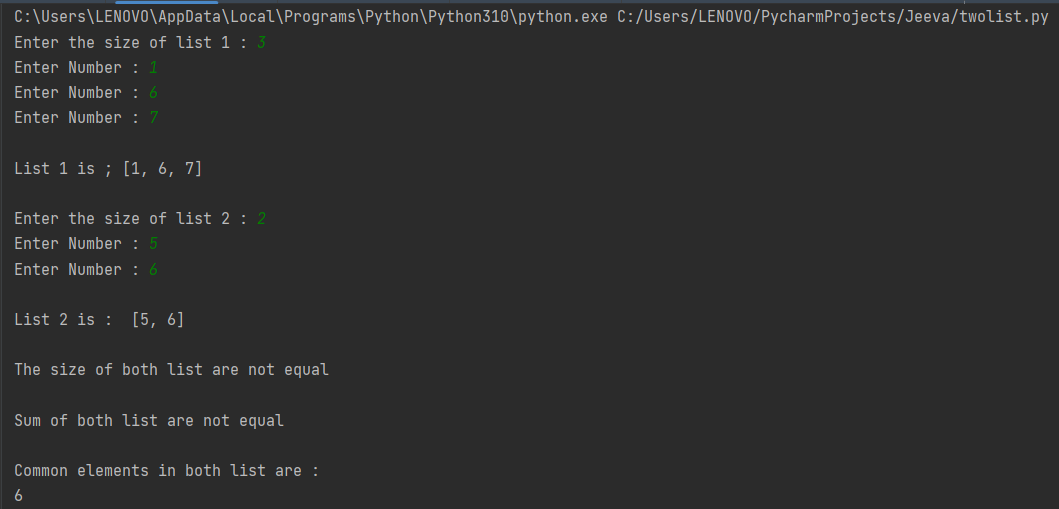
for i in l2:

for j in l4:

if(i==j):

print(i)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 44 Date: 04-11-2022**

**Aim:** Count the number of characters in a string.

**CO2:** Implement decision making, looping constructs and functions.

**Procedure:**

n=str(input("Enter the String: "))

s= {}

for i in n:

if i in s:

s[i]= s[i]+1

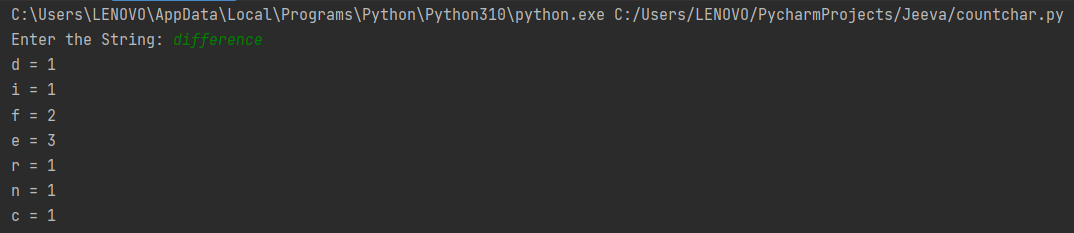
else:

s[i]=1

for m,s in s.items():

print(m ,"=",s)

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**Experiment No.: 45 Date: 04-11-2022**

**Aim:** Find the longest word and find its length.

**CO2:** Implement decision making, looping constructs and functions.

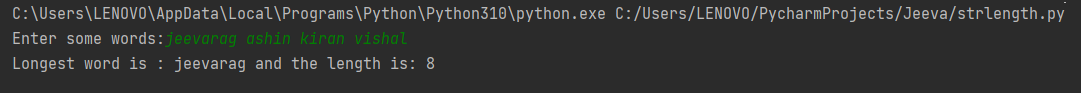
**Procedure:**

sen=input("Enter some words:")

longest=max(sen.split(),key=len)

print("Longest word is :",longest,"and the length is:",len(longest))

**Output Screenshot:**

****

**Result**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**EXPERIMENT 46:**

**AIM:** Add ‘ing’ at the end of a given string. If it already ends with ‘ing’, then add ‘ly’.

**CO:** Implement decision making, looping constructs and functions.

**Procedure:**

st = str(input("Enter a string:"))

if st[-3:] == 'ing':

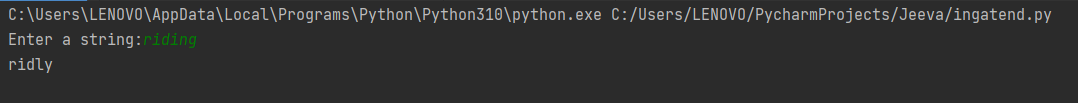
st = st.replace("ing","ly")

else :

st = st + "ing"

print(st)

**Output Screenshot:**



**Result:**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**EXPERIMENT 47:**

**AIM:** Count the occurrences of each word in a line of text.

**CO:** Implement decision making, looping constructs and functions.

**Procedure:**

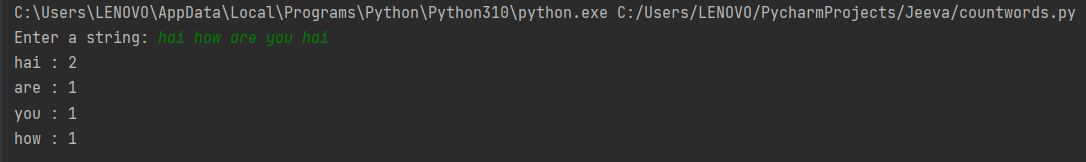
sen = input("Enter a string:").split()

l = (set(sen))

for i in l:

print(i,":",sen.count(i))

**Output Screenshot:**



**Result:**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**EXPERIMENT 48:**

**AIM:** Get a string from an input string where all occurrences of first character replaced with

‘$’, except for the first character.

**CO:** Implement decision making, looping constructs and functions.

**Procedure:**

st1 = str(input("Enter the string:"))

a = st1[0]

for i in st1:

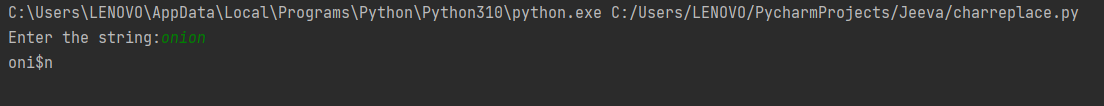
if i==a:

st1 = st1.replace(i,"$")

st1 = a + st1[1:]

print(st1)

**Output Screenshot:**



**Result:**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**EXPERIMENT 49:**

**AIM:** Store a list of first names. Count the occurrences of ‘a’ within the list.

**CO:** Implement decision making, looping constructs and functions.

**Procedure:**

n = int(input("Enter the limit:"))

li= []

count = 0

for i in range (0,n):

fn = input("Enter the first name:")

li.append(fn)

print(li)

for i in li:

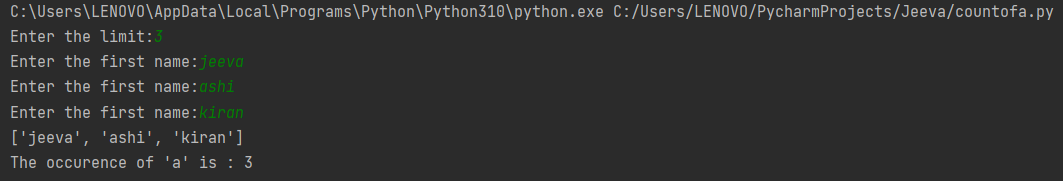
for j in i:

if j=='a':

count = count +1

print("The occurrence of 'a' is :", count)

**Output Screenshot:**



**Result:**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**EXPERIMENT 50:**

**AIM:**  Using Lambda function find the area of the rectangle, square and triangle.

**CO:** Implement decision making, looping constructs and functions.

**Procedure:**

l = int(input("Enter the length of rectangle:"))

b = int(input("Enter the breadth:"))

arec = lambda l, b : l\*b

print("Area of rectangle =",arec(l,b))

s = int(input("Enter the side of square:"))

arsq = lambda s : s\*s

print("Area of square =",arsq(s))

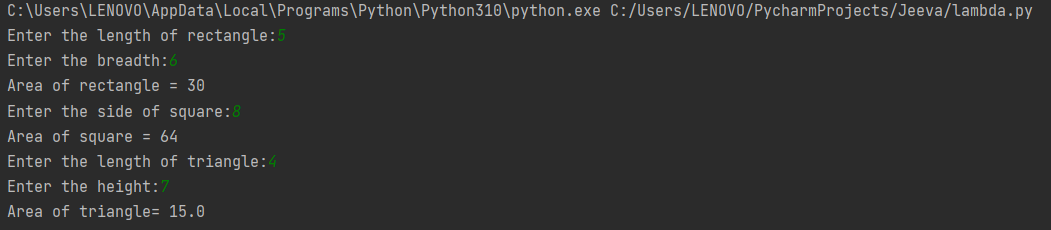
l1 = int(input("Enter the length of triangle:"))

h = int(input("Enter the height:"))

artr = lambda l1, h : 1/2\*l1\*h

print("Area of triangle=",artr(l1,h))

**Output Screenshot:**



**Result:**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**EXPERIMENT 51:**

**AIM:** Find factorial of a number using a function.

**CO:** Implement decision making, looping constructs and functions.

**Procedure:**

n = int(input("Enter the number:"))

def facto(a):

fac = 1

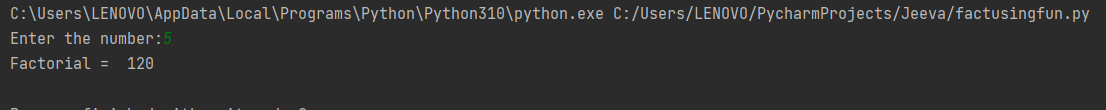
for i in range(2,a+1):

fac = fac\*i

return fac

print("Factorial = ", facto(n))

**Output Screenshot:**



**Result:**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**EXPERIMENT 52:**

**AIM:** Check palindrome using a for loop.

**CO:** Implement decision making, looping constructs and functions.

**Procedure:**

num=int(input("Enter the number:"))

rev=0

num1=num

for i in range(num):

if num != 0:

r=num%10

num=num//10

rev=rev\*10+r

else:

break

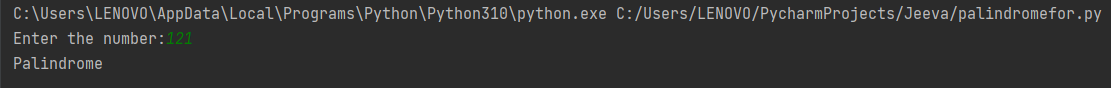
if rev == num1:

print("Palindrome")

else:

print("Not Palindrome")

**Output Screenshot:**



**Result:**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.

**EXPERIMENT 53:**

**AIM:** Reverse of a number using a for loop.

**CO:** Implement decision making, looping constructs and functions.

**Procedure:**

num=int(input("Enter the number:"))

rev=0

num1=num

for i in range(num):

if num != 0:

r=num%10

num=num//10

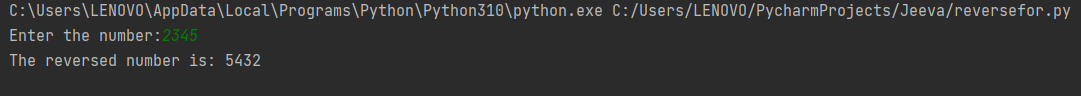
rev=rev\*10+r

else:

break

print("The reversed number is:",rev)

**Output Screenshot:**



**Result:**

The program was executed and the result was successfully obtained. Thus, CO2 was obtained.