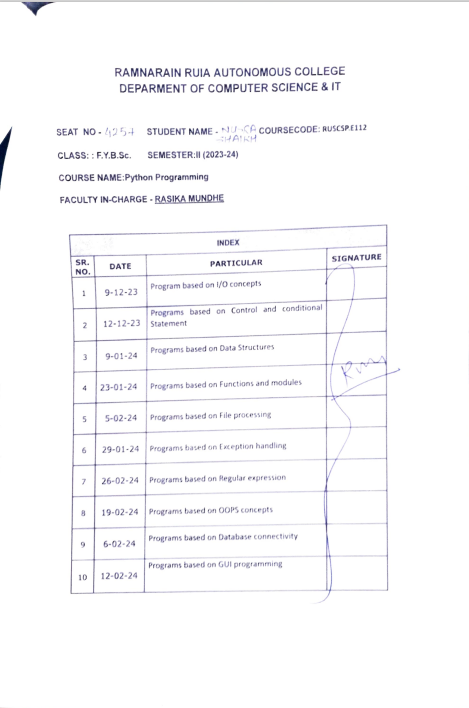
**Python**





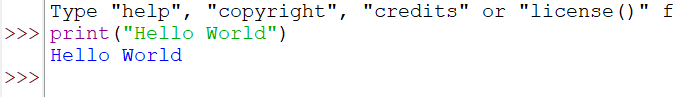
**Practical 1:Programs based on I/O concepts.**

1. WRITE A PYTHON PROGRAM TO PRINT “hello world”.

Code:

Print(“hello  world”)

Output:

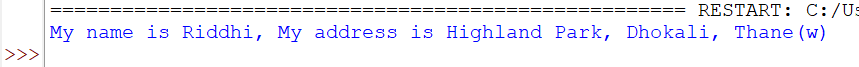


1. WRITE A PYTHON PROGRAM TO PRINT YOUR NAME AND ADDRESS

Code:

print("My name is Riddhi, My address is Highland Park, Dhokali, Thane(w)")

Output:



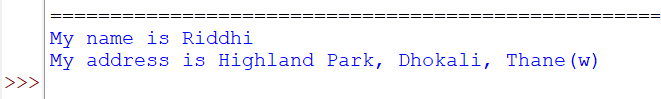
3. WRITE A PYTHON PROGRAM TO PRINT YOUR NAME AND ADDRESS      oN A MULTILINE.

Code:

print("My name is Riddhi")

print("My address is Highland Park, Dhokali, Thane(w)")

Output:



4.WRITE A PYTHON PROGRAM TO PRINT THE FOLLOWING PATTERN.

Code:

print("\*\*\*\*\*\*\*\*\*\*#")

print("\*\*\*\*\*\*\*\*\*#")

print("\*\*\*\*\*\*\*\*#")

print("\*\*\*\*\*\*\*#")

print("\*\*\*\*\*\*#")

print("\*\*\*\*\*#")

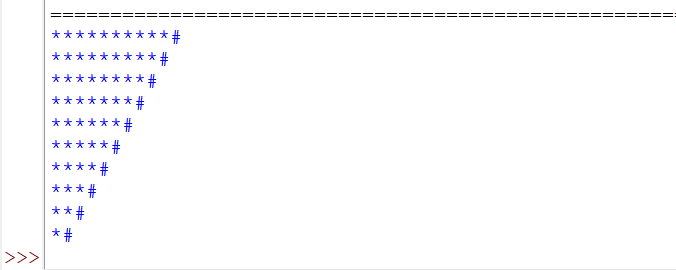
print("\*\*\*\*#")

print("\*\*\*#")

print("\*\*#")

print("\*#")

Output:



**Programs using input**

Q1: Write a python program to find the average of three numbers

Code:

a=int(input("enter 1st number"))

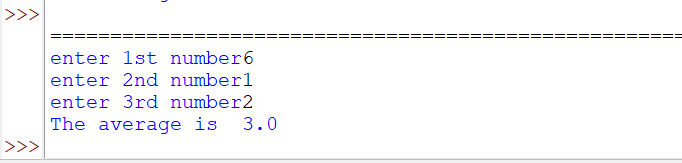
b=int(input("enter 2nd number"))

c=int(input("enter 3rd number"))

avg=(a+b+c)/3

print("The average is ",avg)

Output:



Q2:Write a program to convert the distane from KM to meters

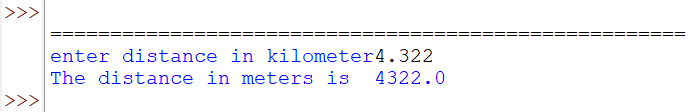
Code:

a=float(input("enter distance in kilometer"))

m=a\*1000

print("The distance in meters is ",m)

Output:



Q3:Write a python program to convert time from hours to seconds:

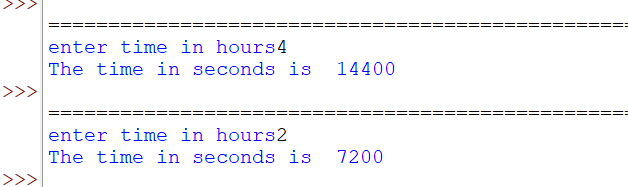
Code:

hours=int(input("enter time in hours"))

seconds = hours\*60\*60

print("The time in seconds is ",seconds)

Output:



Q4:Write a python program to accept length,breadth and height of a

box and print its area and volume

Code:

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

b=int(input("enter breadth"))

h=int(input("enter height"))

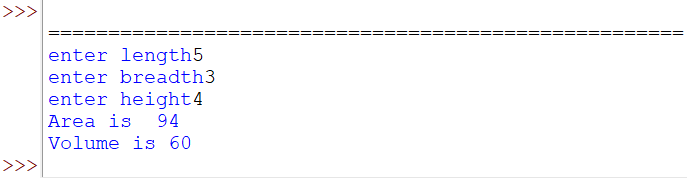
area=2\*(l\*b+b\*h+h\*l)

volume=l\*b\*h

print("Area is ", area)

print("Volume is", volume)

Output:



Q5:Write a python program to calculate the total distance travelled by a vehical in t seconds is given by &quot;distance=ut+(at)^2\*1/2&quot;

Code:

n1=int(input("enter velocity"))

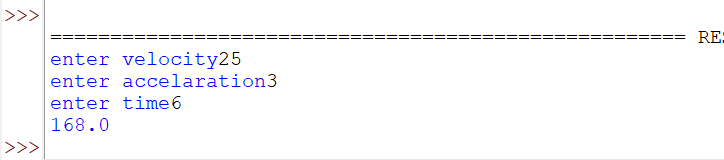
n2=int(input("enter accelaration"))

n3=int(input("enter time"))

distance=n1\*n3+0.5\*n2\*n3\*2

print(distance)

Output:



Q6:Write a python to print a number and its square:

Code:

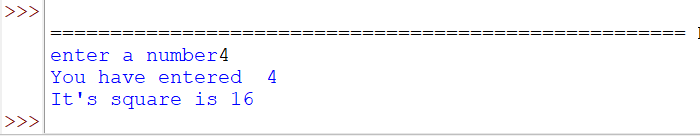
n=int(input("enter a number"))

s=n\*n

print("You have entered ",n)

print("It's square is",s)

Output:



Q7:Wite a python program to find the sum, subtraction, division and

product of two numbers.

Code:

n1=int(input("enter 1st number"))

n2=int(input("enter 2nd number"))

add=n1+n2

sub=n1-n2

product=n1\*n2

div=n1/n2

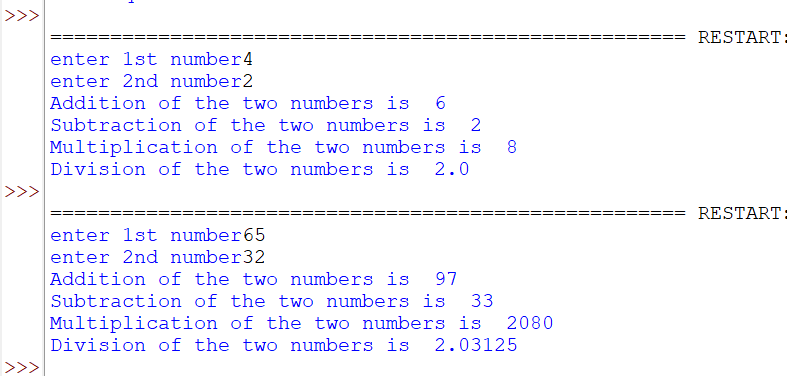
print("Addition of the two numbers is ", add)

print("Subtraction of the two numbers is ", sub)

print("Multiplication of the two numbers is ", product)

print("Division of the two numbers is ", div)

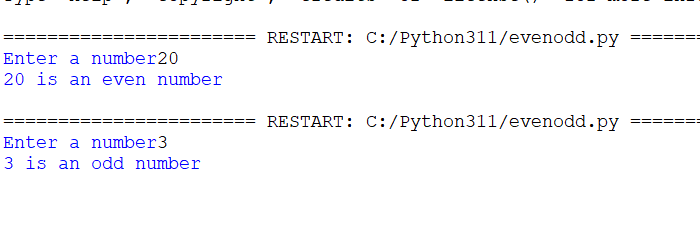
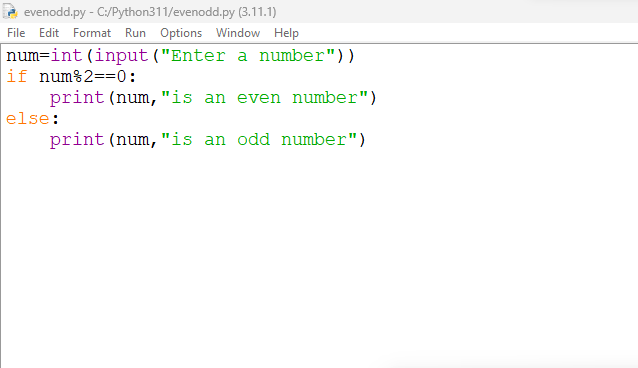
Output:



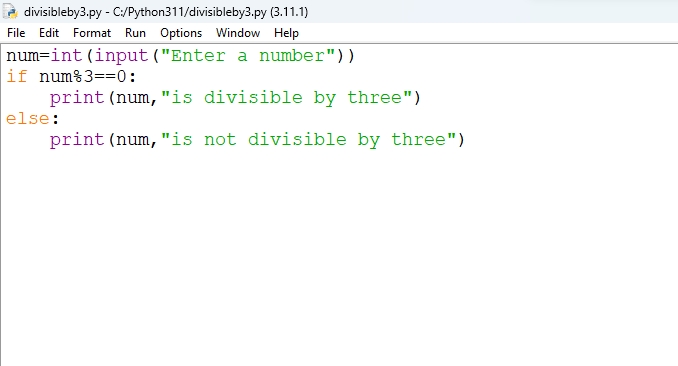
Practical-2:Programs based on control and conditional statement.

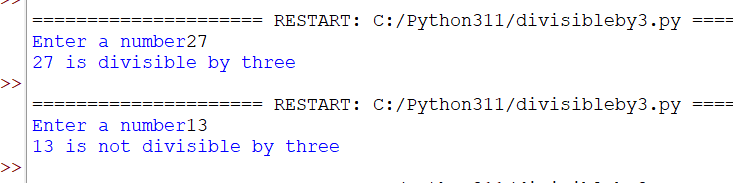
(IF/ELSE)

1. Write a Python program to check whether the number entered is even or odd.

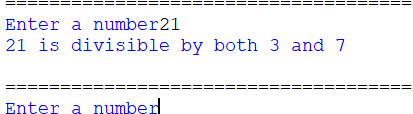
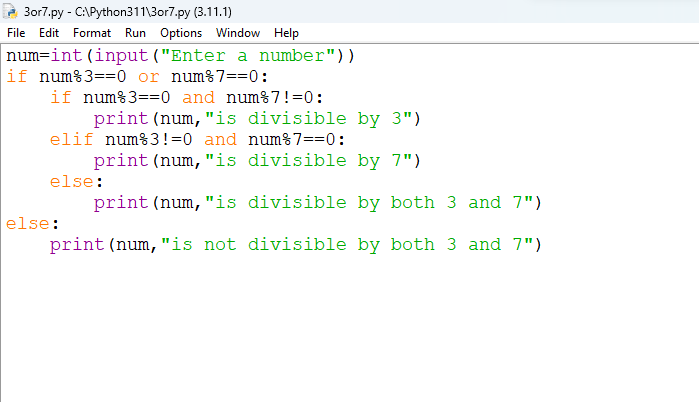


1. Check whether the number entered is divisible by 3 or not.

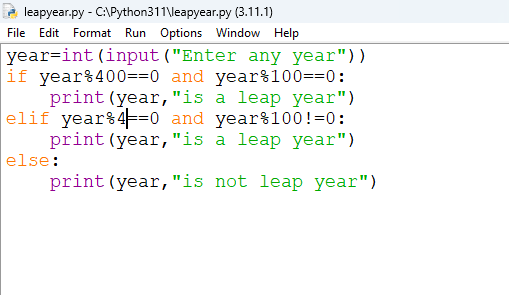


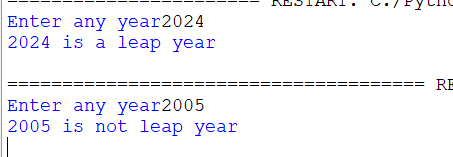


1. Write a program o check whether the number is divisible by 3 or 7.

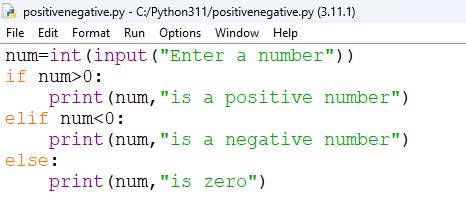


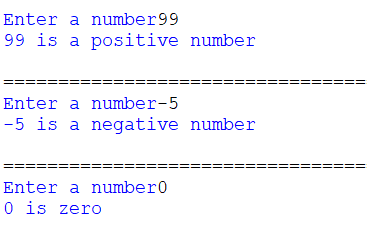
1. Wap to enter a 4-digit number and check whether it is leap year or not.



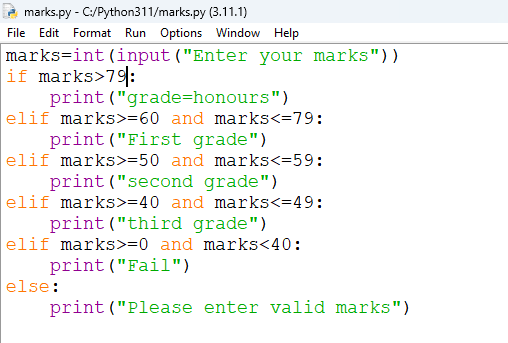


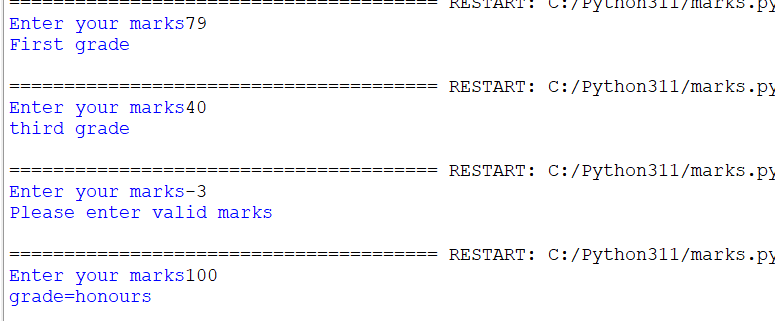
1. Wap to check whether the number entered is positive,negative or zero.



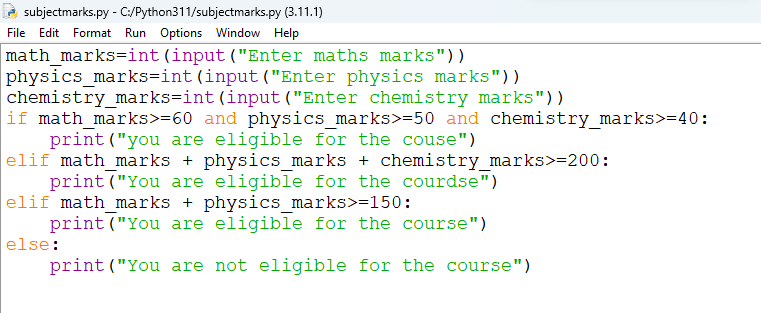


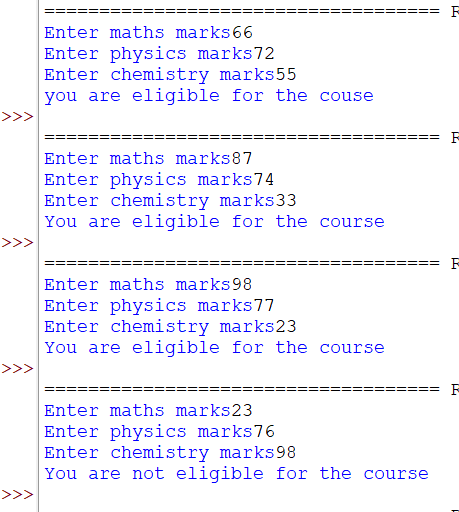
1. Wap to grade the students in an academic institution according to the following groups:



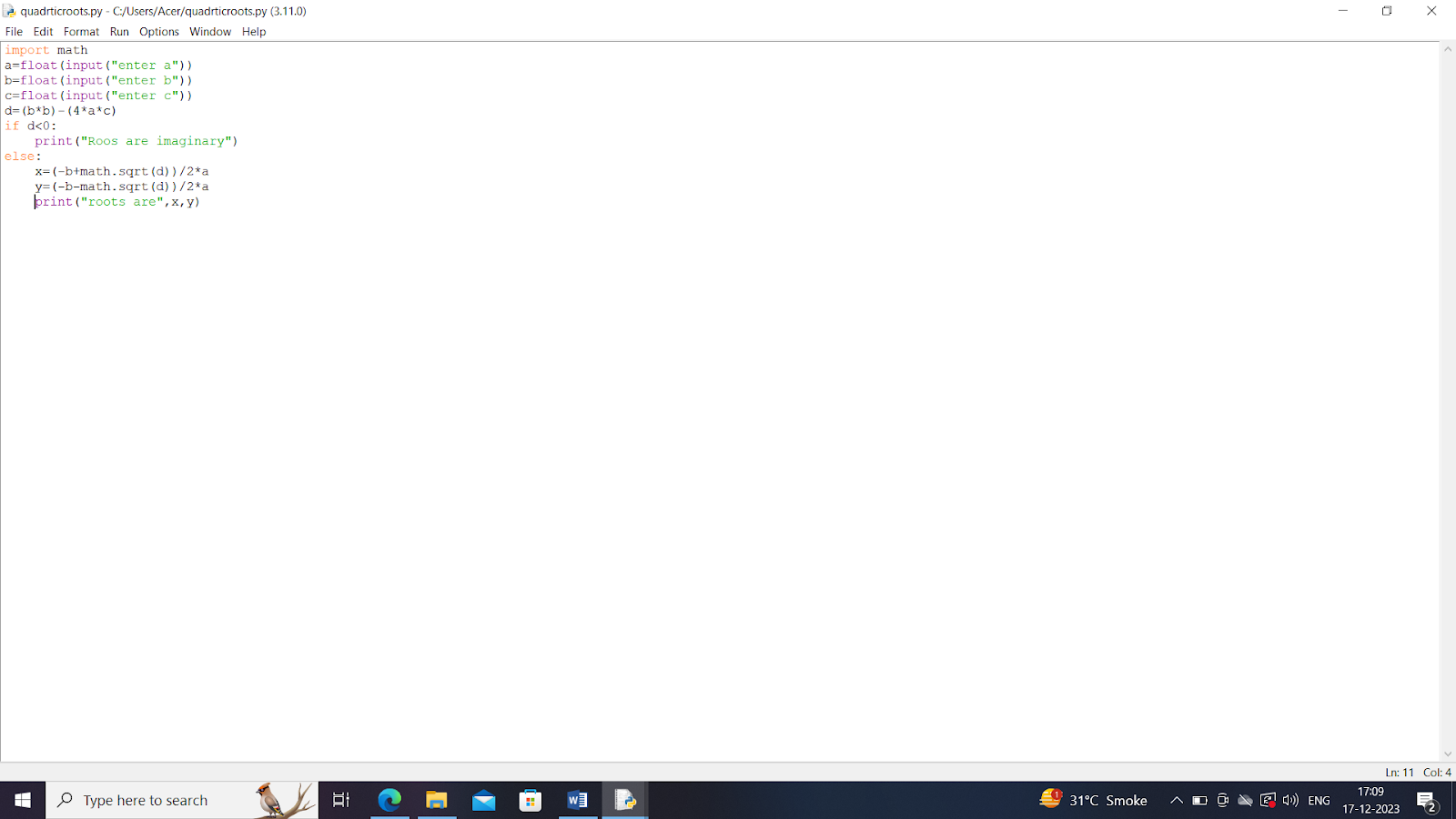


1. Admission to a professional course is subject to the following conditions.



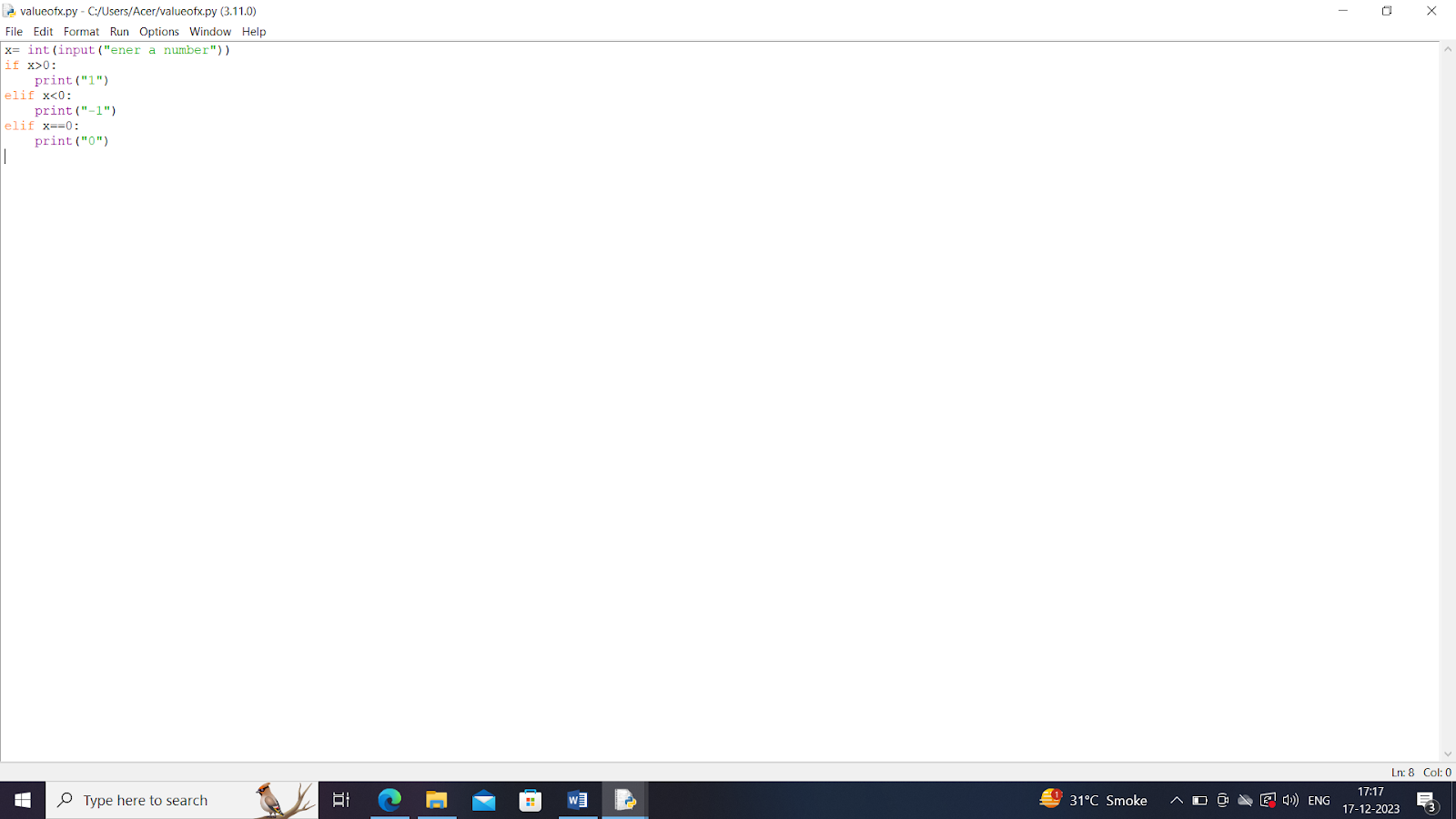


1. Wap to compute the real roots of a quadratic equation.



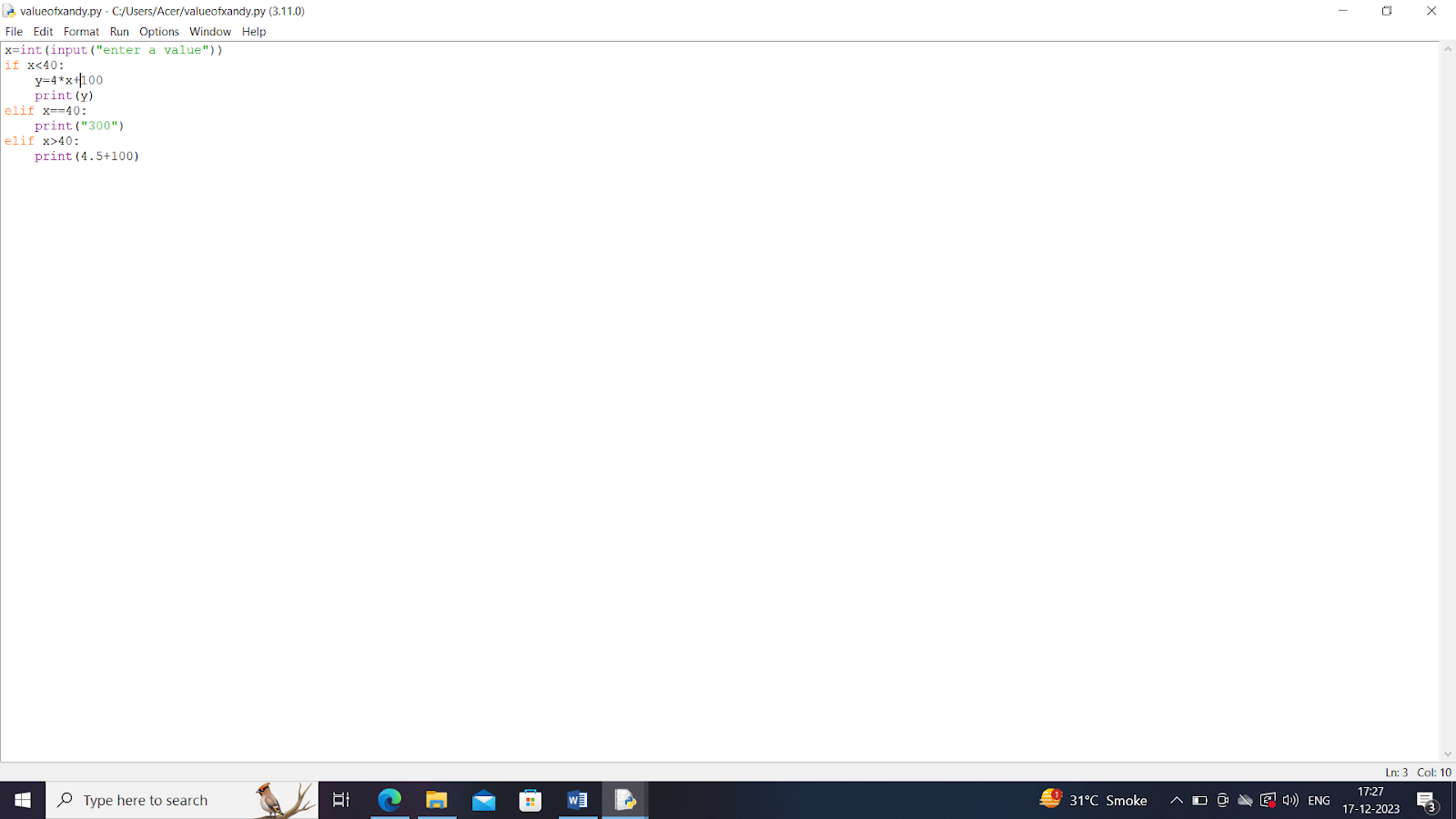


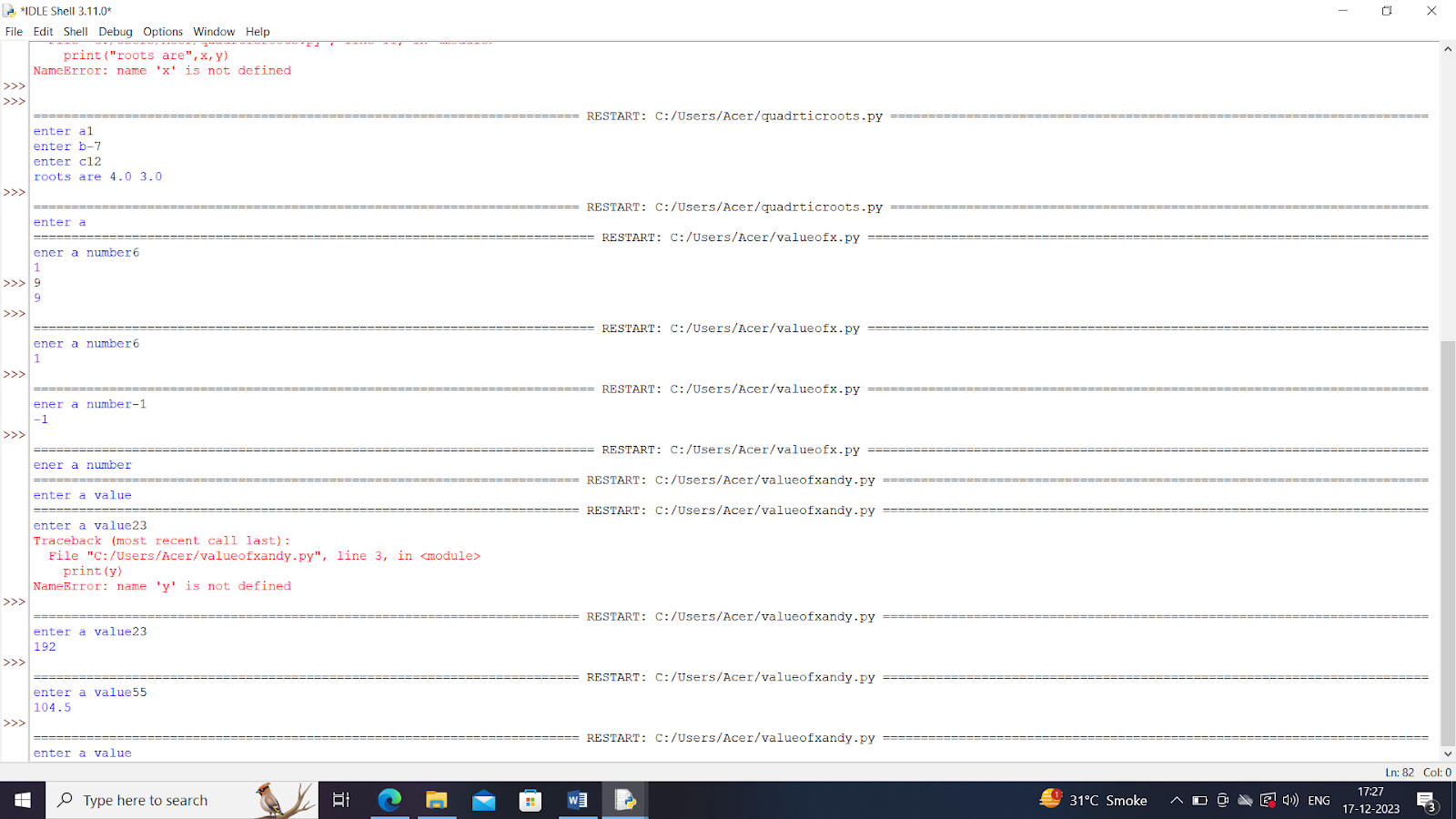
1. Wap that will read the value of x evaluate the following function y=1 for x>0 0 for x=0 -1for x<0.





1. Wap that will read the value of x and evaluate the following function.





**Loops**

**1.Program to accept a number from user and print its digits.**

Code:

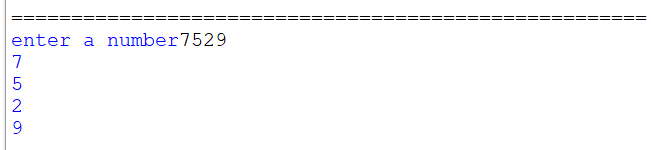
num=int(input("enter a number"))

s=str(num)

for i in s:

    print(i)

Output:



**2.Program to perform addition and subtraction of two numbers until the user wishes**.

Code:

choice="y"

while choice=="y" :

    no1=int(input("enter 1st no."))

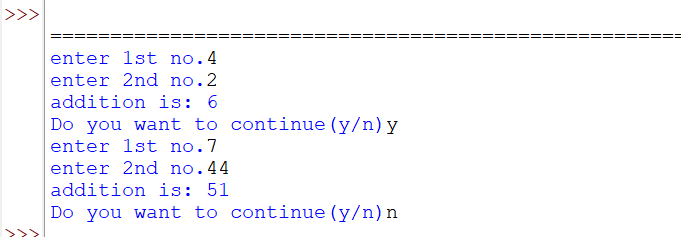
    no2=int(input("enter 2nd no."))

    res=no1+no2

print("addition is:",res)

    choice=input("Do you want to continue(y/n)")

Output:



**3.Program to calculate factorial of a number**

Using for loop:

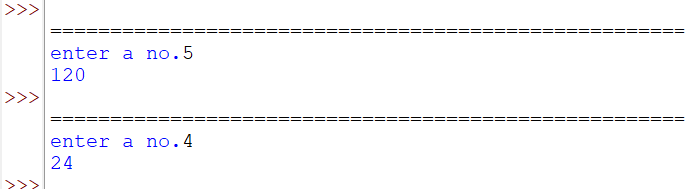
n=int(input("enter a no."))

for i in range(1, n):

 n=n\*i

print(n)

Output:



Using while loop:

n=int(input("enter a no."))

fact=1

i=1

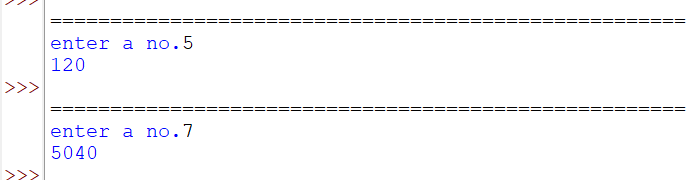
while i<=n:

 fact=fact\*i

i=i+1

print(fact)

Output:



**4.Program to perform the sum of first 10 natural numbers**

Using while loop:

sum=0

num=1

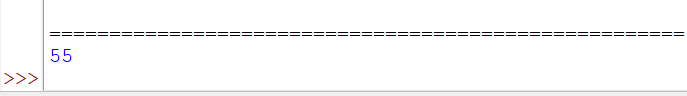
while (num<=10):

    sum=sum+num

    num=num+1

print(sum)

Output:



Using for loop:

sum=0

num=1

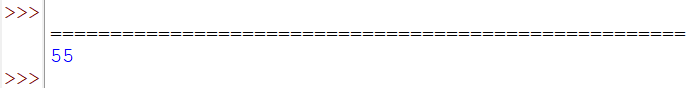
for i in range(1,11):

    sum=sum+num

    num=num+1

print(sum)

Output:



**5.Program to check whether the number entered is  Armstrong or not.**

Code:

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

temp=num

sum=0

while temp > 0:

    digit = temp % 10

    sum += digit \*\* 3

    temp //= 10

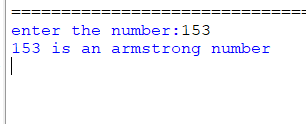
if num == sum:

print(num,"is an armstrong number")

else:

print(num,"is NOT an armstrong number")

Output:



**6.Programs to display factors of a number.**

Code:

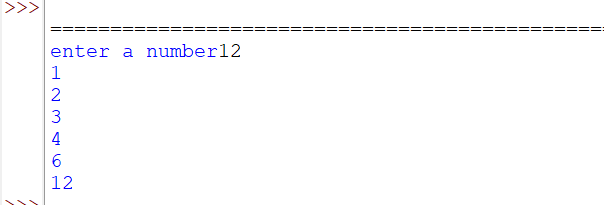
n=int(input("enter a number"))

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

    if n%i==0:

        print(i)

Output:



**7.Program to check whether the number is perfect or not.**

Code:

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

sum=0

for i in range(1,n):

    if(n%i==0):

        sum=sum+i

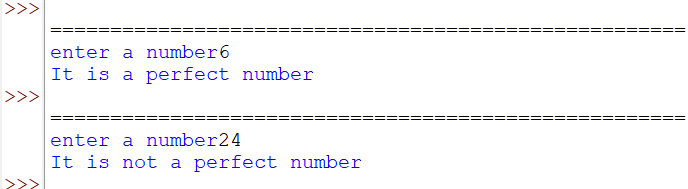
if(sum==n):

print("It is a perfect number")

else:

print("It is not a perfect number")

Output:



**8.Program to accept a number and print Fibonacci series.**

Code:

t1=0

t2=1

nextterm=t1+t2

n=int(input('enter number of terms'))

print('fibonacci series:',t1,t2)

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

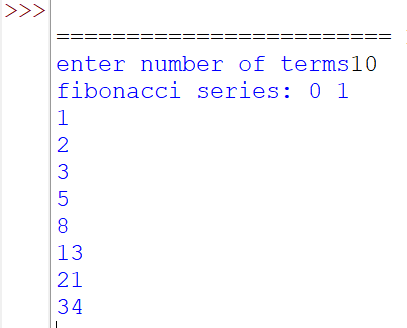
    print(nextterm)

    t1=t2

    t2=nextterm

nextterm=t1+t2

Output:



**9.Program to reverse a number.**

Code:

num=int(input('Enter a number to reverse:'))

reverse=0

while num>0:

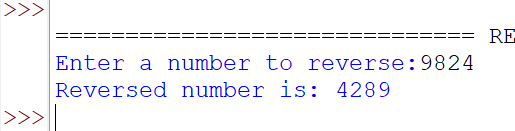
    digit=num%10

    reverse=(reverse\*10)+digit

num=num//10

print('Reversed number is:',reverse)

Output:



**10. Program to check whether the number is a palindrome or not.**

Code:

num = int(input("Enter a number to check whether it is palindrom or not: "))

reverse = 0

while num > 0:

    digit = num % 10

    reverse = reverse \* 10 + digit

    num = num // 10

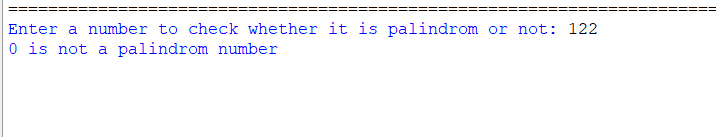
if num == reverse:

print(num, "is a palindrom number")

else:

print(num, "is not a palindrom number")

Ouput:



**11. Program to print prime numbers between 1 to 100.**

Code:

print('prime numbers between 1 to 100 are')

for num in range(2,101):

    for i in range(2,num):

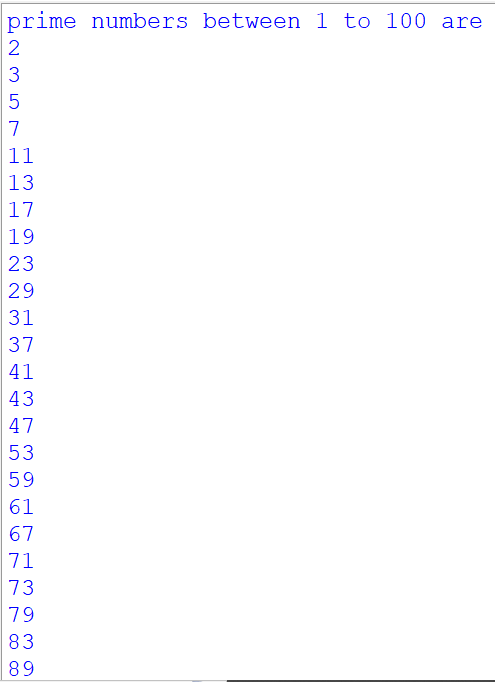
        if (num%i==0):

            break

    else:

        print(num)

Output:





**12.Program to perform addition and multiplication of all odd and even numbers between 1 to 100.**

Code:

odd\_sum = 0

even\_sum = 0

odd\_pro = 1

even\_pro = 1

for num in range(1, 101):

    if num%2 == 0:

even\_sum += num

even\_pro \*= num

    else:

odd\_sum += num

odd\_pro \*= num

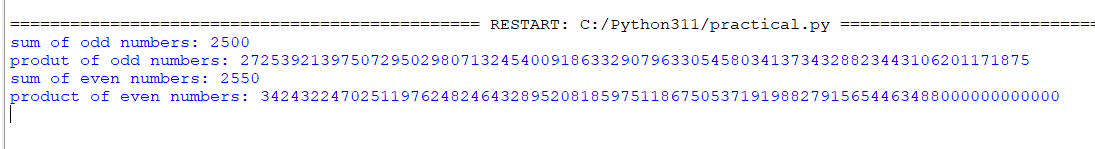
print("sum of odd numbers:", odd\_sum)

print("produt of odd numbers:", odd\_pro)

print("sum of even numbers:",even\_sum)

print("product of even numbers:",even\_pro)

Output:



**13.Program to print all prime numbers till 1 to 30.**

Code:

prime\_sum=0

for num in range(1,31):

    if num>1:

        for i in range(2,num):

            if (num%i)==0:

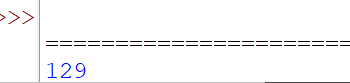
                break

        else:

prime\_sum = prime\_sum+num

print(prime\_sum)

Output:



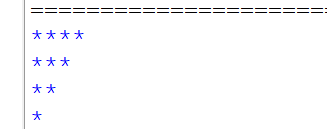
**14.Printing star pattern.**

Code:

for i in range(4,0,-1):

    print("\*"\*i)

Output:



**15. Program to print sum of digits of given number**

Code:

num=int(input('enter a number'))

sumOfDigits=0

while num>0:

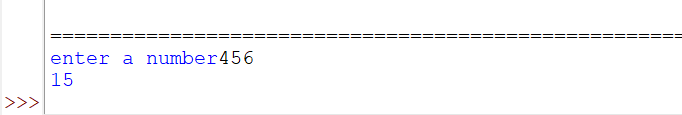
    digit=num%10

    sumOfDigits += digit

    num//=10

print(sumOfDigits)

Output:

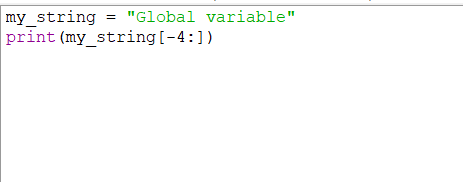


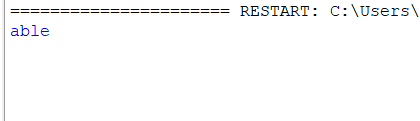
Practical 3: Programs on data structures.

**Strings**

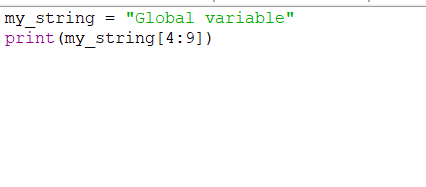
***1) Consider the string “Global variable” & Write statements in python to implement the following:***

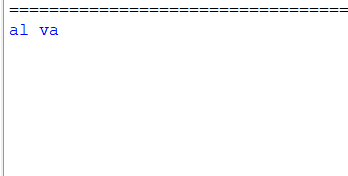
*a. To display the last four characters:*

**

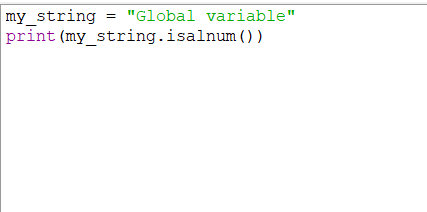
**

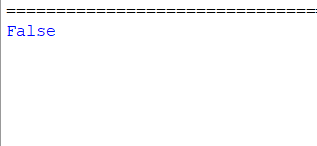
*b. To display the substring starting from index 4 and ending at 8:*

**

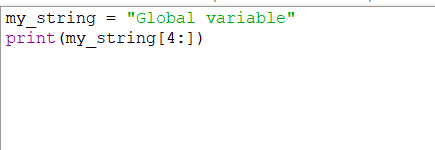
**

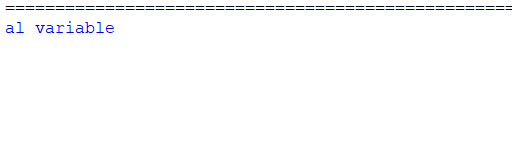
*c. To check whether string has alphanumeric character or not.our character from the string:*

**

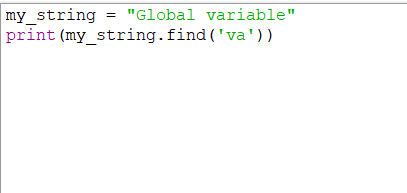
**

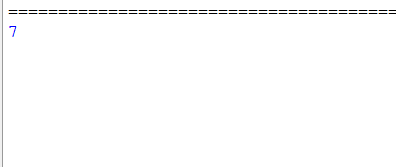
*e. To trim the first four letter of the string:*

**

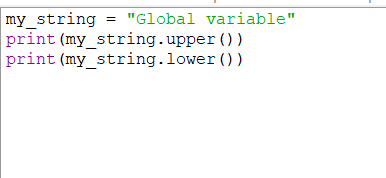
**

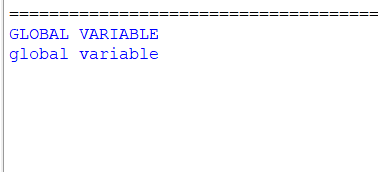
*f. To display starting index from the substring “va”:*

**

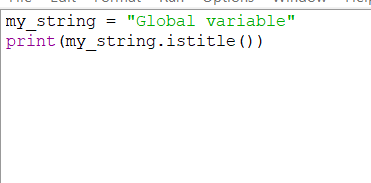
**

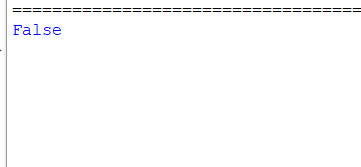
*g. To change the case of the given string:*

**

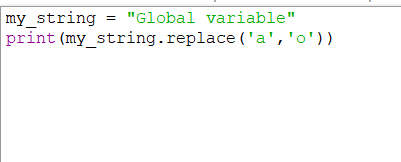
**

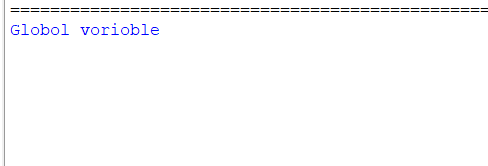
*h. To check whether the string is in title case:*

**

**

*i. To change all the occurences of letter ‘a’ with ‘o’:*

**

**

***2.* *Wap in python tp check whether the entered number is palindrome or not.***

num = int(input("Enter a number to check whether it is palindrom or not: "))

num\_str = str(num)

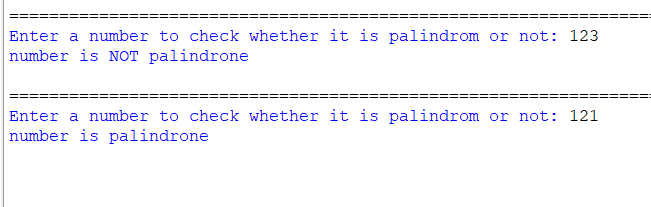
rev = num\_str[::-1]

if (rev == num\_str):

    print("number is palindrone")

else:

    print("number is NOT palindrone")

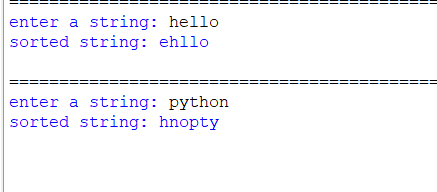


***3. Wap in python to sort the string entered.***

string = input("enter a string: ")

sorted = ''.join(sorted(string))

print("sorted string:",sorted)



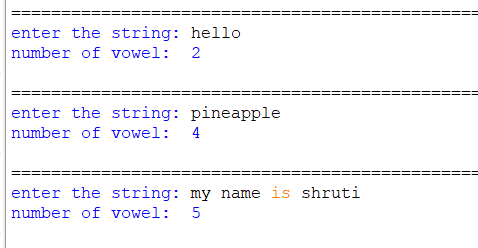
***4. Wap in python to count the number of vowels in a string.***

string = input("enter the string: ")

lower = string.lower()

vowel = sum(lower.count(vowel) for vowel in "aeiou")

print("number of vowel: ",vowel)



***5.program to accept a string from the user and cheks it starts with a or not.***

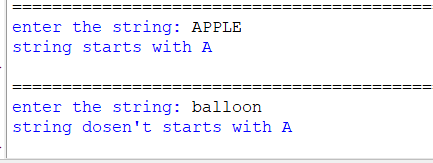
string = input("enter the string: ")

if string.startswith('A') :

    print("string starts with A")

else :

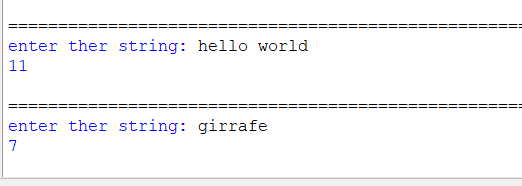
    print("string dosen't starts with A")



***6. Write a python program to calculate the length of a string.***

string = input("enter ther string: ")

print(len(string))



***7.wap to count the number of characters(character frequency) in a string.***

user\_input = input("Enter a string: ")

character\_frequency = {}

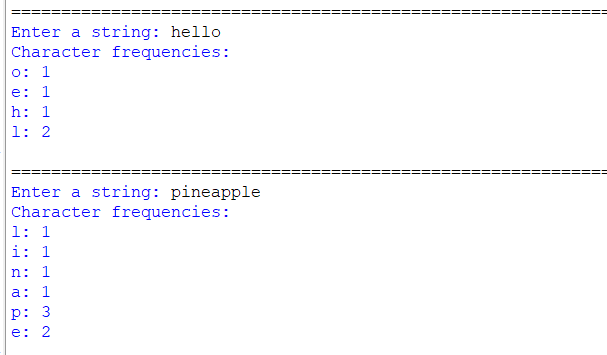
for char in set(user\_input):

    character\_frequency[char] = user\_input.count(char)

print("Character frequencies:")

for char, count in character\_frequency.items():

    print(f"{char}: {count}")

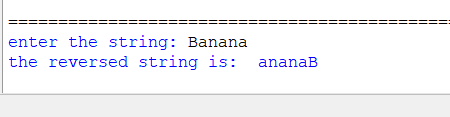


***8.Write a python program to reverse a string and string is “Banana”.***

string = input("enter the string: ")

rev = string[::-1]

print("the reversed string is: ",rev)



***9. Write a python program to get a string made of first 2 and last 2chars from a given string.if str length is less than 2 ,return instead of the empty string.***

user\_input = input("Enter a string: ")

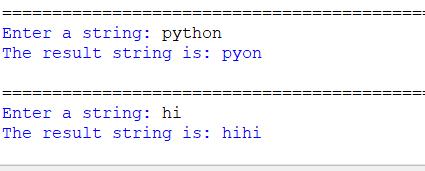
if len(user\_input) < 2:

    result\_string = ""

else:

    result\_string = user\_input[:2] + user\_input[-2:]

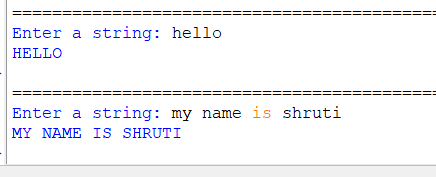
print(f"The result string is: {result\_string}")



***10.wap that takes input from the user and displays that input back in upper***:

string = input("Enter a string: ")

print(string.upper())

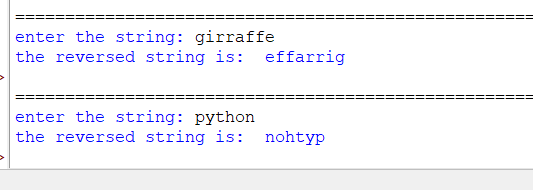


***11. Wap in python to reverse the entered string.***

string = input("enter the string: ")

rev = string[::-1]

print("the reversed string is: ",rev)

**

***12. Write a python program to add “ing” at the end of the given string.If the given string already ends with “ing” add “ly” instead. If the string length of the given string is less than 3 leave it unchanged.***

string = input("enter the string: ")

if len(string) < 3 :

    result = string

else :

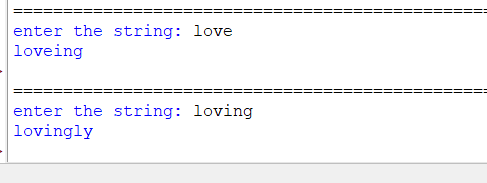
    if string.endswith("ing") :

        result = string + "ly"

    else :

        result = string + "ing"

print(result)



***13.write a python program to change a given string to new string by exchanging first and the last letter.***

string = input("enter the string: ")

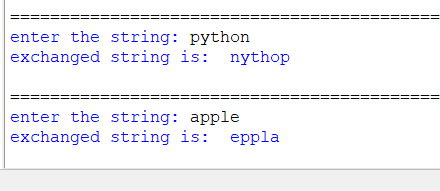
if len(string) >= 2:

    result = string[-1] + string[1:-1] + string[0]

else:

    result = string

print("exchanged string is: ",result)



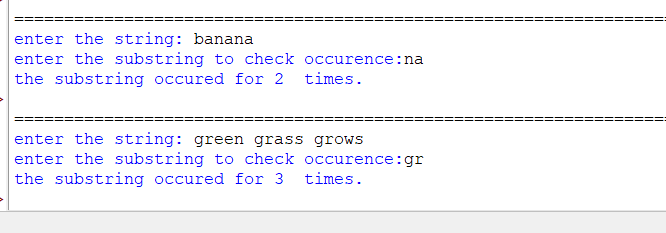
***14. Write a python program to count occurrence of substring.***

string = input("enter the string: ")

sub = input("enter the substring to check occurence:")

occurence = string.count(sub)

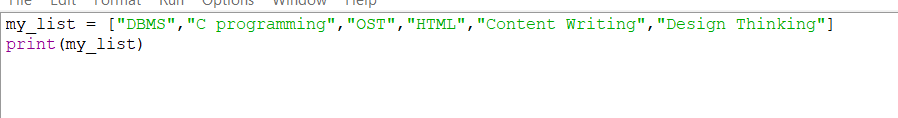
print("the substring occured for",occurence," times.")



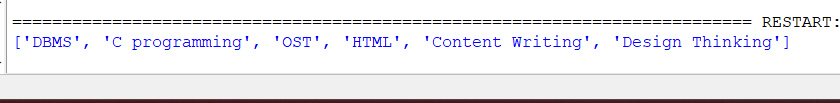
**LISTS**

**1. WAP to create a list of semester 1 subject and display it.**

**CODE:**

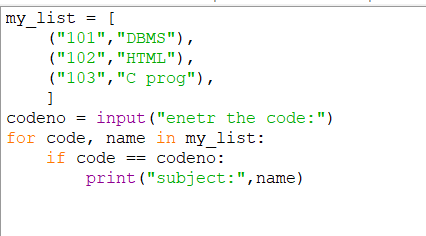


**OUTPUT:**

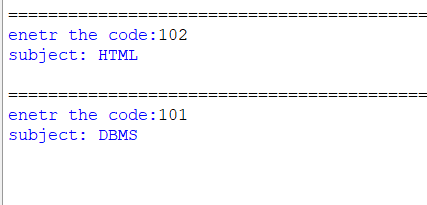


**2.WAP ask student to input subject code and display subject name**

**CODE:**

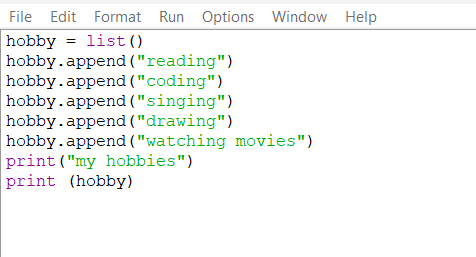
****

**OUTPUT:**

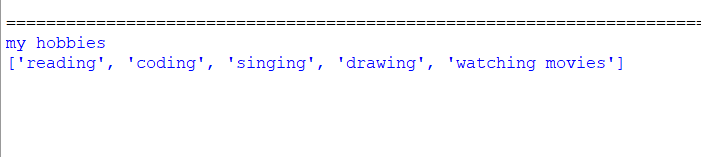
****

**3. Create a empty list to set your targets and add any five targets depends on priority.**

**CODE:**

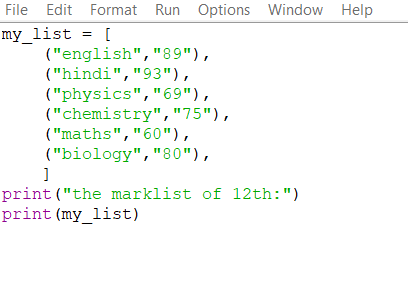
****

**OUTPUT:**

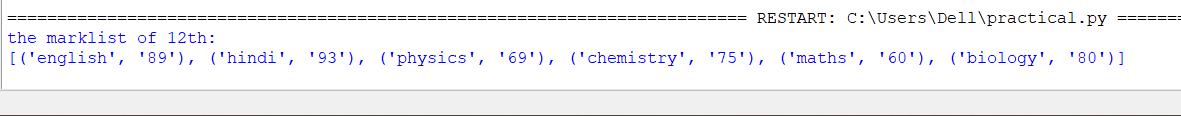
****

**4.WAP to create a list of your 12th std Marks**

**CODE:**

****

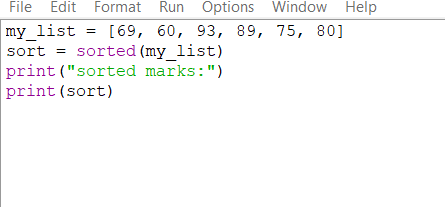
**OUTPUT:**

****

**5. Perform the following operation in the above list:**

**a. Generate a new list(in descending order of your marks)**

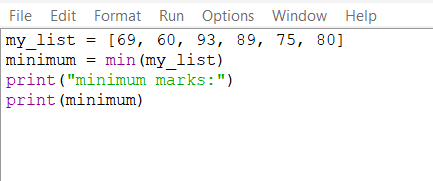
**CODE:**

****

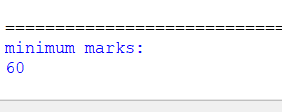
**OUTPUT:**

**b.Display min marks in 12th.**

**CODE:**

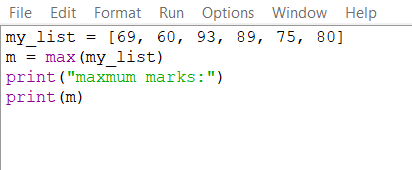
****

**OUTPUT:**

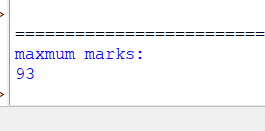
****

**C. Display max marks in 12th.**

**CODE:**

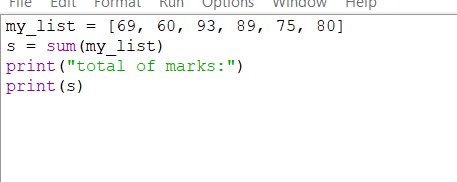
****

**OUTPUT:**

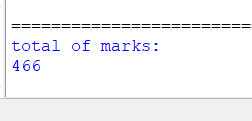
****

**d.Display total marks.**

**CODE:**

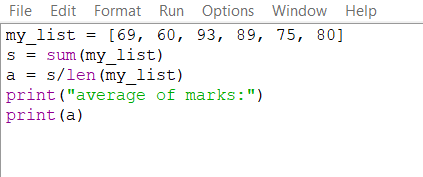
****

**OUTPUT:**

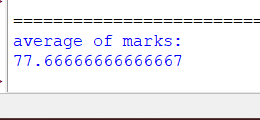
****

**E. Display average marks.**

**CODE:**

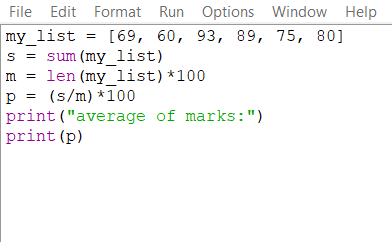
****

**OUTPUT:**

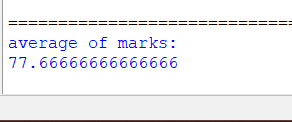
****

**F. Calculate percentage.**

**CODE:**

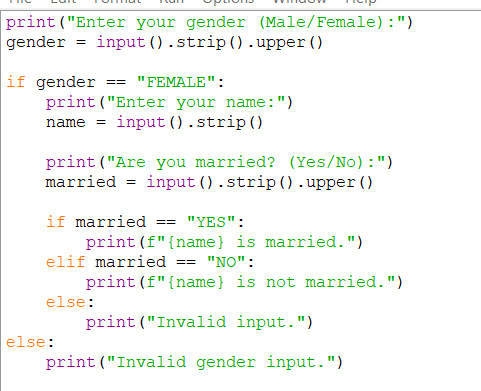
****

**OUTPUT:**

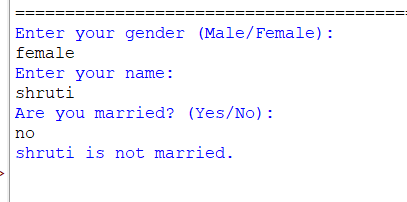
****

**6. Accept a number from consumer along with prefix (female) & check if she is married or not.**

**Code:**

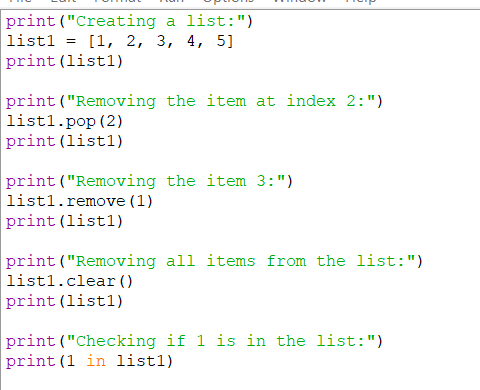
****

**Output:**

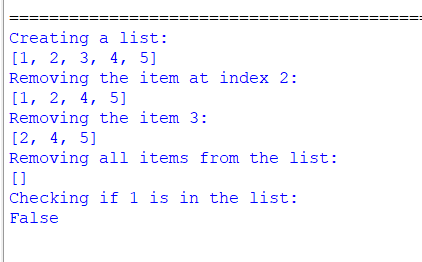
****

**7.Create a list and demonstrate delete methods of Lists.**

**Code:**

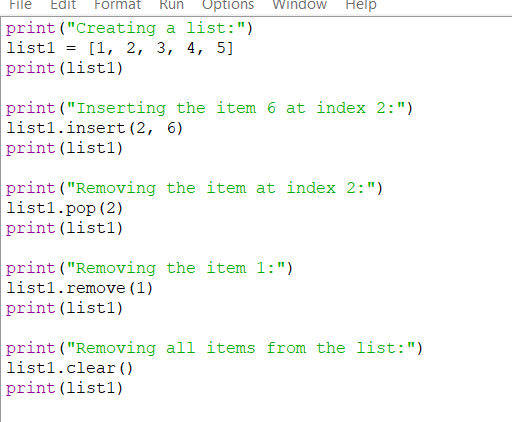
****

**Output:**

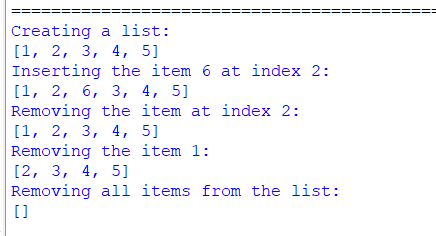
****

**8.Create a list and insert an element at a specific place.**

**Code:**

****

**Output:**

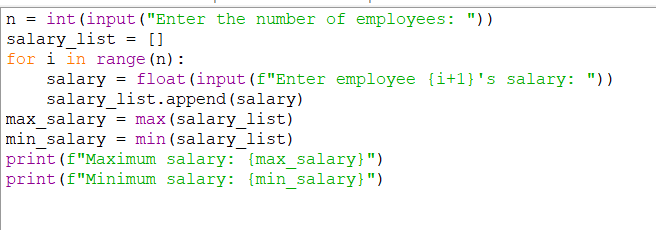
****

**1.write a python program to input &#39;n&#39; employee`s salary**

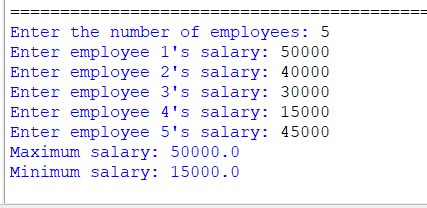
**and find maximum and minimum salary among &#39;n&#39;**

**employees.**

**Code:**

****

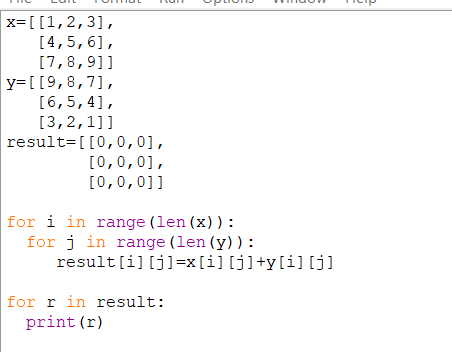
**Output:**

****

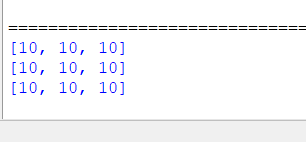
**2.write a python program to accept two input n x m and**

**find sum**

**code:**

****

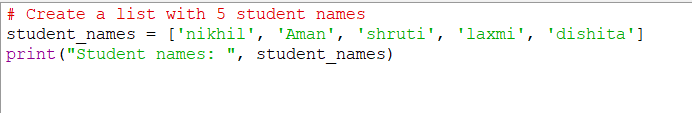
**Output:**

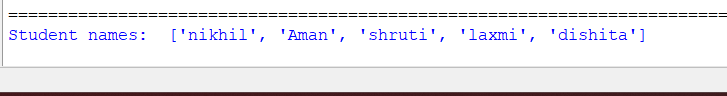
****

**3. Create a list that contains the names of 5 students of**

**your class.**

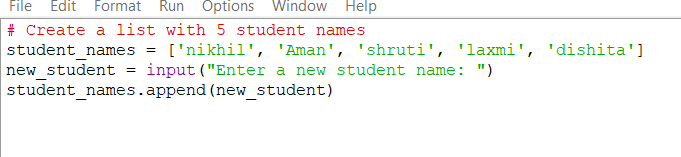
**a.print the list**

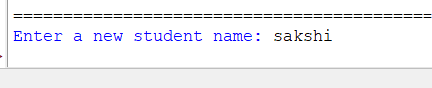
****

****

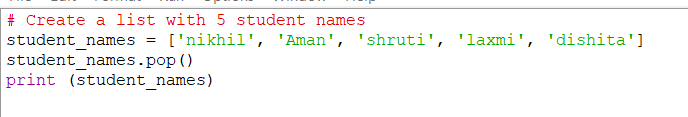
**b.ask the user to input one name and append it to the**

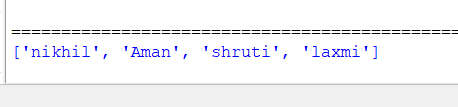
**list**

****

****

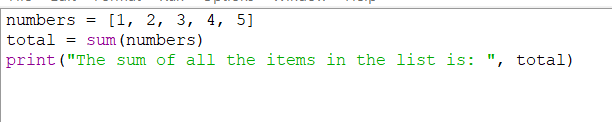
**c.remove the last element of the list**

****

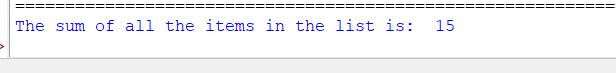
****

**4.WAP TO SUM ALL THE ITEMS IN THE LIST.**

**Code:**

****

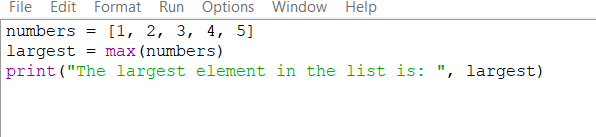
**Output:**

****

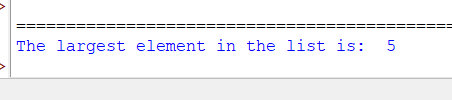
**5.WAP TO TO DISPLAY THE LARGEST ELEMENT IN THE**

**LIST.**

**Code:**

****

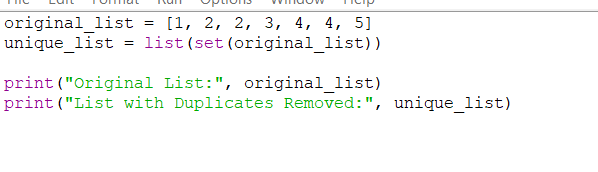
**Output:**

****

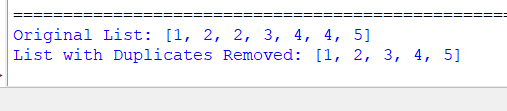
**6.write a python program to remove duplicates from the**

**list**

**code:**

****

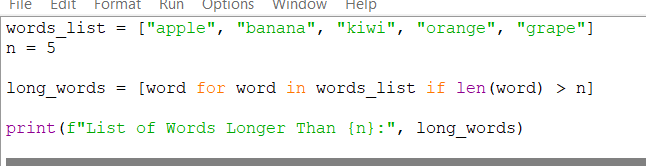
**Output:**

****

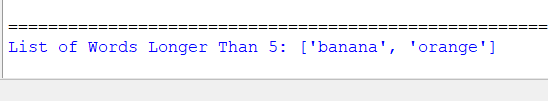
**7.write a python program to find the list of words that**

**are longer than n from a given list of words**

**code:**

****

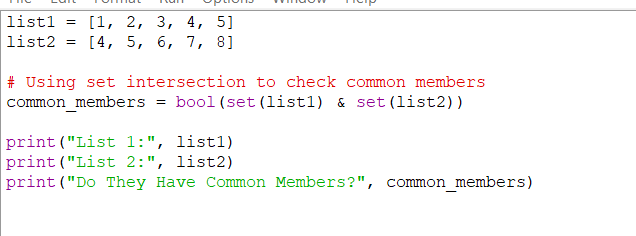
**Output:**

****

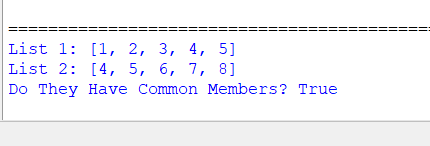
**8.write a python program that takes two lists and returns**

**true if they have at least one common member**

**code:**

****

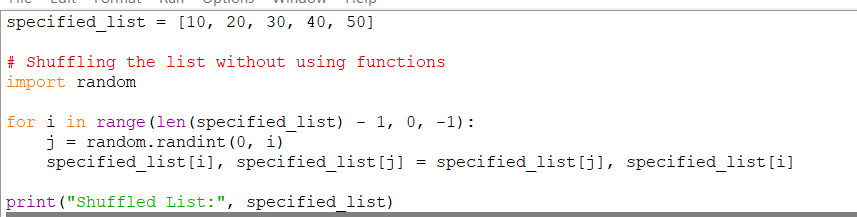
**Output:**

****

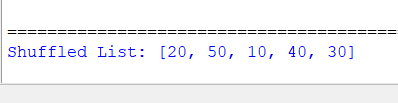
**9.WRITE A PYTHON PROGRAM TO SHUFFLE AND PRINT**

**THE SPECIFIED LIST.**

**Code:**

****

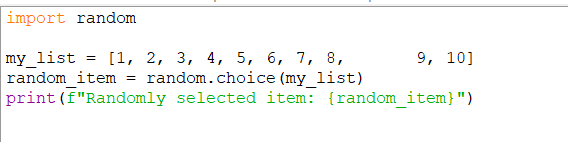
**Output:**

****

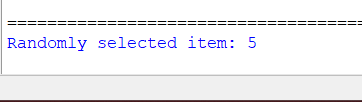
**10.WRITE A PYTHON PROGRAM TO SELECT AN ITEM**

**RANDOMLY FROM LIST.**

**Code:**

****

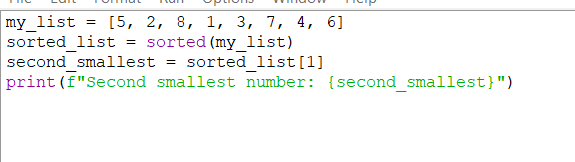
**Output:**

****

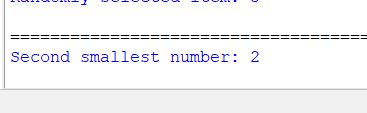
**11.WRITE A PROGRAM TO FIND THE SECOND SMALLEST**

**NUMBER IN THE LIST.**

**Code:**

****

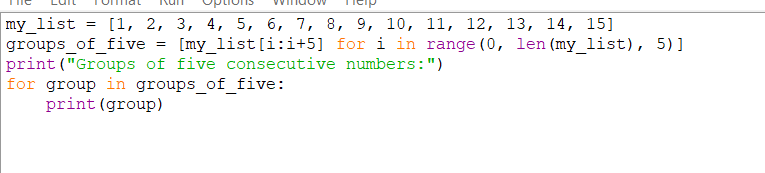
**Output:**

****

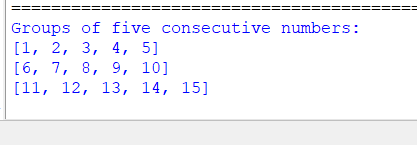
**12.WRITE A PROGRAM TO GENERATE GROUPS OF THE**

**FIVE CONSECUTIVE NUMBERS IN THE LIST.**

**Code:**

****

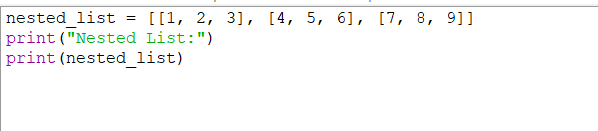
**Output:**

****

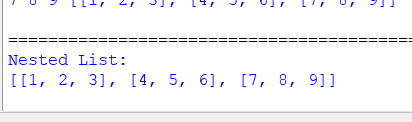
**13.WRITE A PYTHON PROGRAM TO PRINT THE NESTED**

**LIST.**

**Code:**

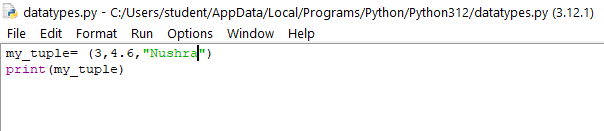
****

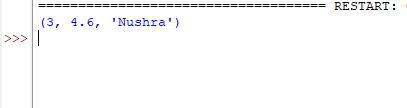
**Output:**

****

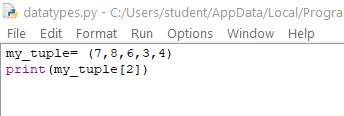
**TUPLES**

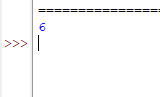
1. Write a python program to create a tuple with different data types.



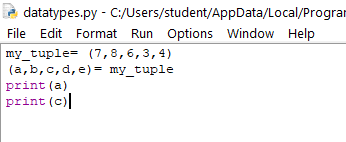


1. Write a python program to create tuple with numbers and print one item.



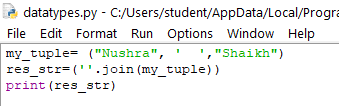


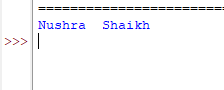
1. Write a python program to create tuple with numbers and print after unpacking the tuple.



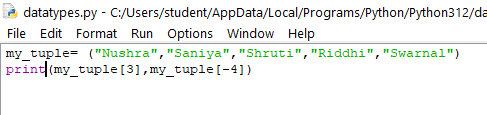


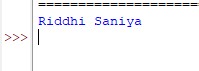
1. Write a python program to convert a tuple to string.



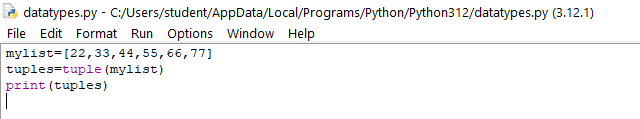


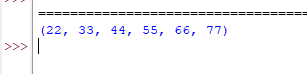
1. Write a python program to get fourth element and last fourth element printed.



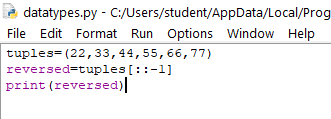


1. Write a python program to convert list to tuple.



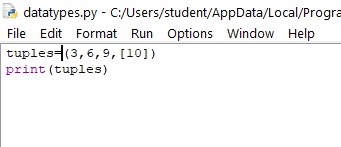


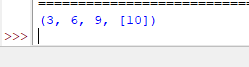
1. Write a python program to print reversed order of tuple.



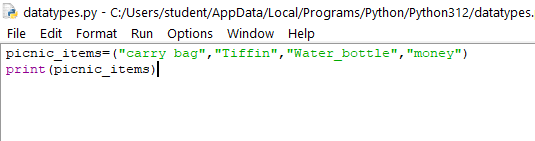
https://lh7-us.googleusercontent.com/NoFAnLrImmYr5KfvM64zyiA_mu04Oc49RiBXzC8j9cm7cf_DOcYAKWEeWdZXBxmTl8D-otUoFiaEq-yDxZqI7c45-oT5XdxZepb1xL0OvbHZWuNYtZr7tPtpdQpxEygeS8nspbI_l9rU3-YSlGFyBsrBrY8zc77y

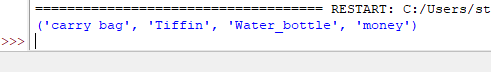
1. Replace last value of tuples in a list.



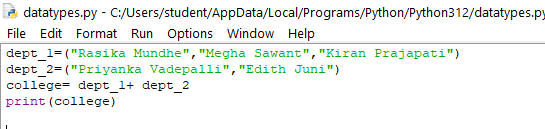


1. Initialize a tuple to print the things needed to bring on the day of picnic.





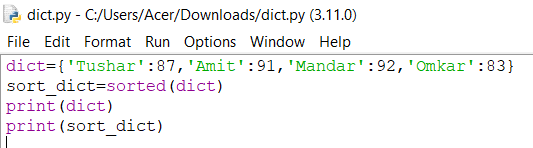
1. Initialize two tuples for name of teachers from two department and then merge it inside college tuple.

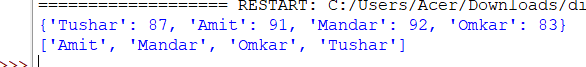


https://lh7-us.googleusercontent.com/X5XtNulTQtko4Ie6qevCO2rmSdUE9ODfgeNNcyfIE-dNblIMM7j5N_58RoqQIVyA-M_GpFWvRxbBPrY-tXW08ittD8XGwM73keUo2dnAjBx0qVmgUuuo9LWX40Hi2q-TYIQG-RhjZvr_eel7wijpQ2P-JWeVMPYt

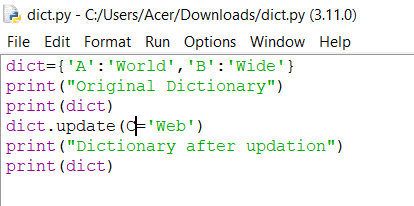
**Dictionary**

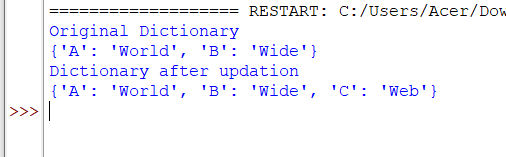
1. Write a python script to sort a dictionary by value.



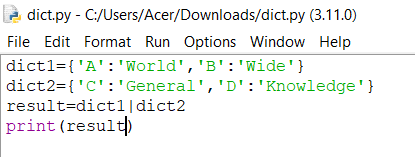


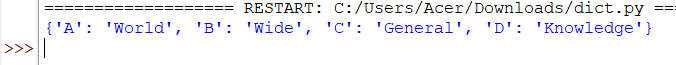
1. Write a python script to update a dictionary.



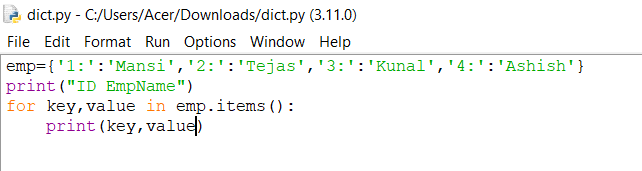


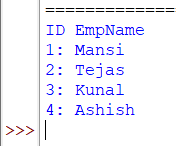
1. Write a python script to concatenate two dictionaries.



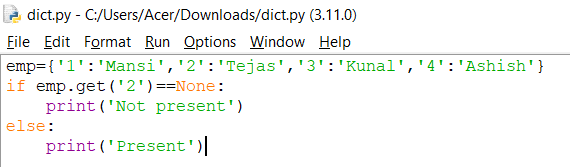


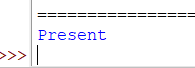
1. Write a python script to print employee id and name.



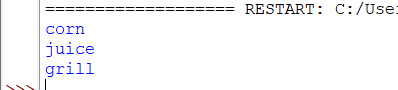


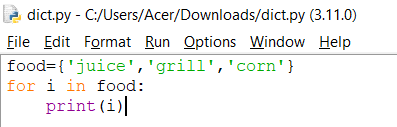
1. Wap in dictionary to check whether the given key exist already.



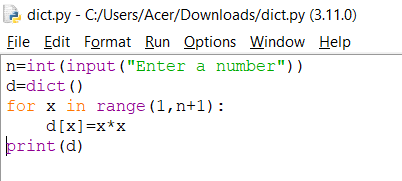


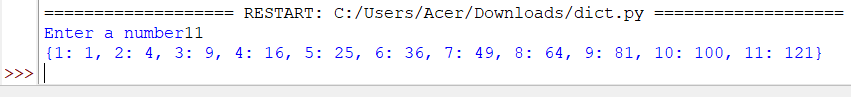
1. Wap in python to iterate over dictionary using for loop.



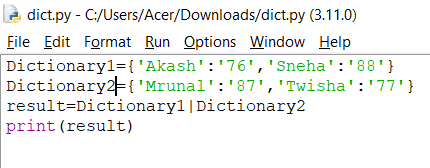


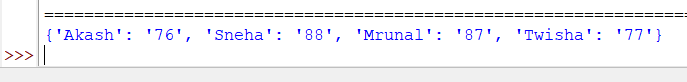
1. Wap script to generate and print a dictionary that contains a number between 1 and x.(,x\*x)



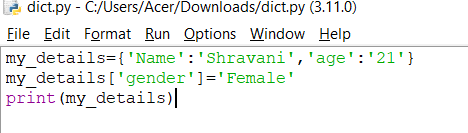
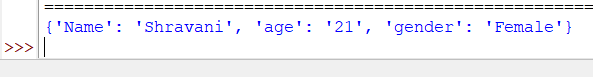


1. Wap script to merge two dictionaries .

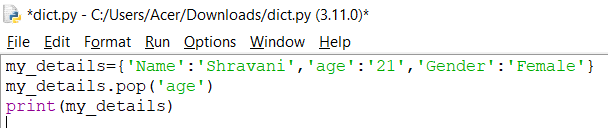


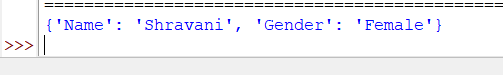


1. Wap in python to add a values in the dictionary.

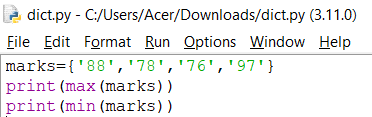
 

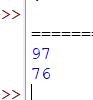
1. Write a python program to remove a key from the dictionary.



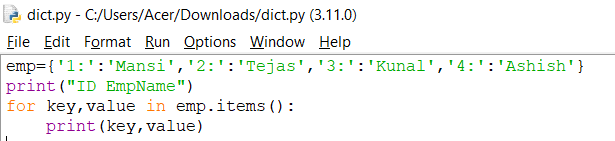


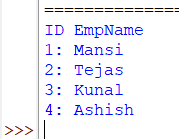
1. Write a python program to print the maximum and minimum values from the dictionary.



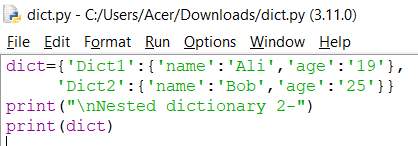


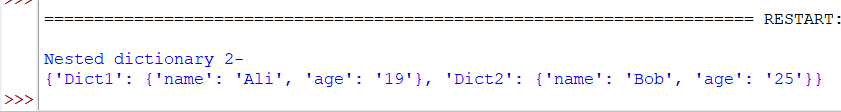
1. Write a python program to print the keys along with the values in a dictionary.



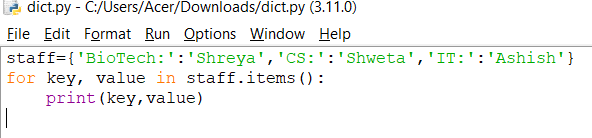


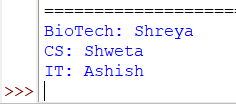
1. Write a python program to print the nested dictionary.





1. Write a python program to print the department as well as the staff in the dictionary.





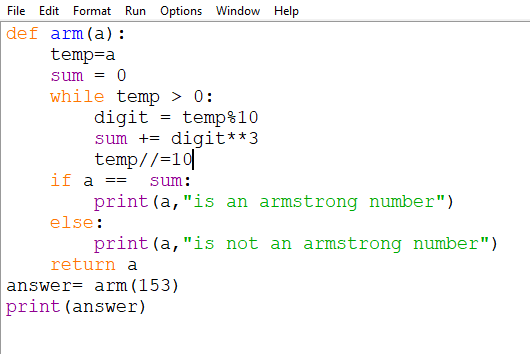
**PRACTICAL 4: FUNCTIONS and MODULES**

**FUNCTIONS**

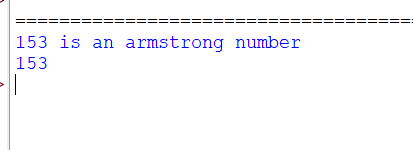
**Q. Write a modular program using functions to do following:**

**1. To check whether number armstrong or not:**

CODE:

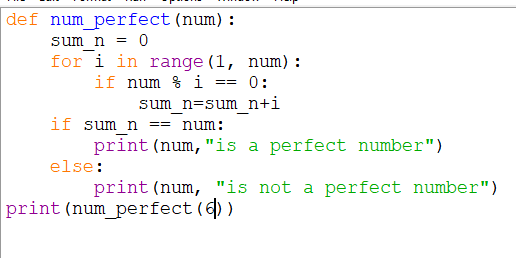


OUTPUT:

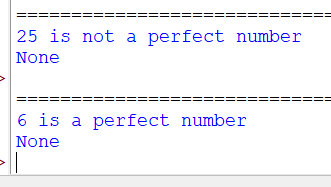


**2. To check whther number perfect or not**:

CODE:

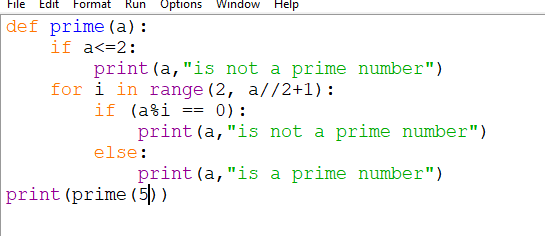


OUTPUT:

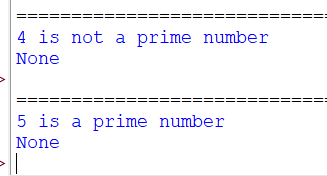


**3. To check prime or not:**

CODE:

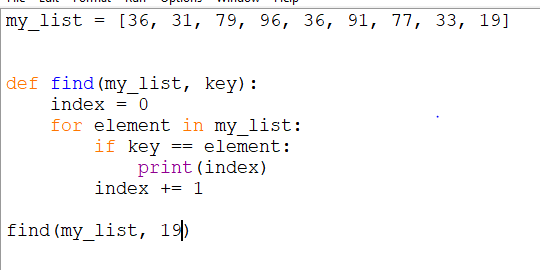


OUTPUT:

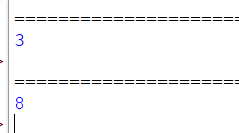


**4. Function to search some element from list:**

CODE:

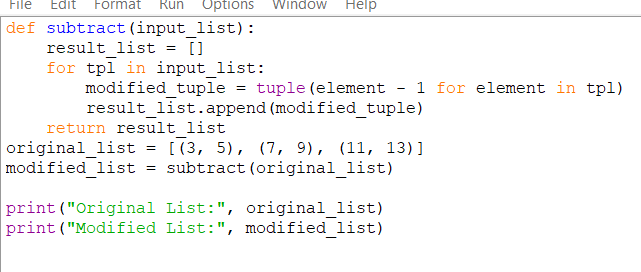


OUTPUT:

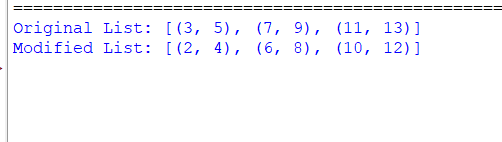


**5. Function to accept tuples from list and subtract 1 from every element**

CODE:

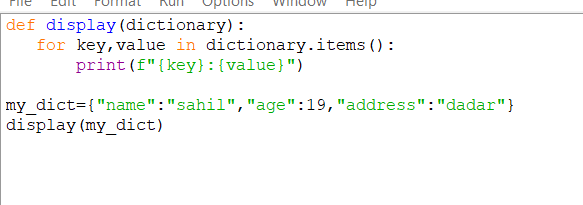


OUTPUT:

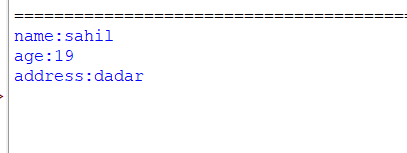


**6. Functions to display contents from dictionary**

CODE:

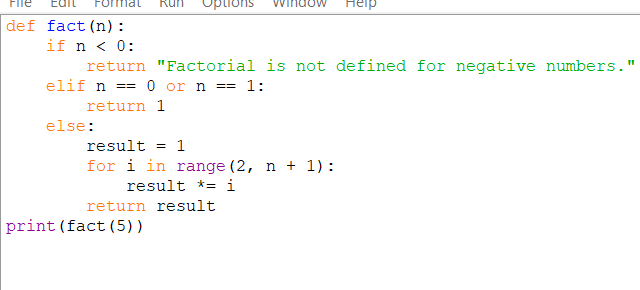


OUTPUT:

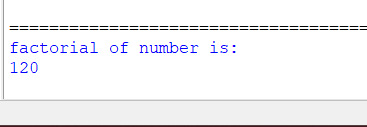


**7. Function to calculate factorial of number**

CODE:

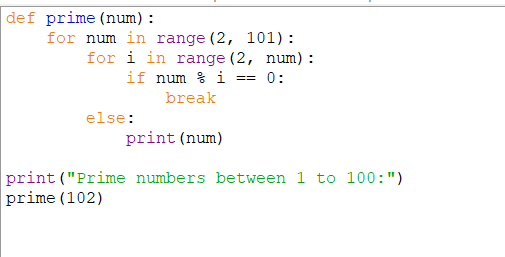


OUTPUT:

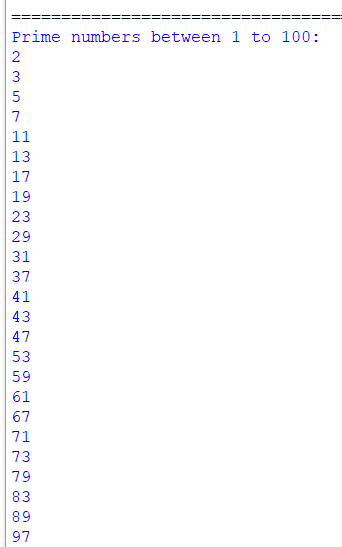


**8. Function to display prime numbers between 1 to 100**

CODE:

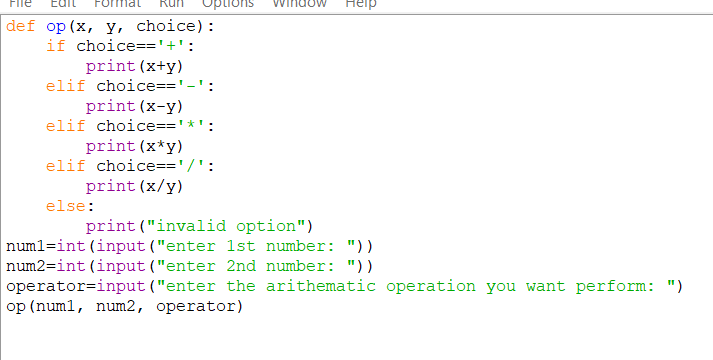


OUTPUT:

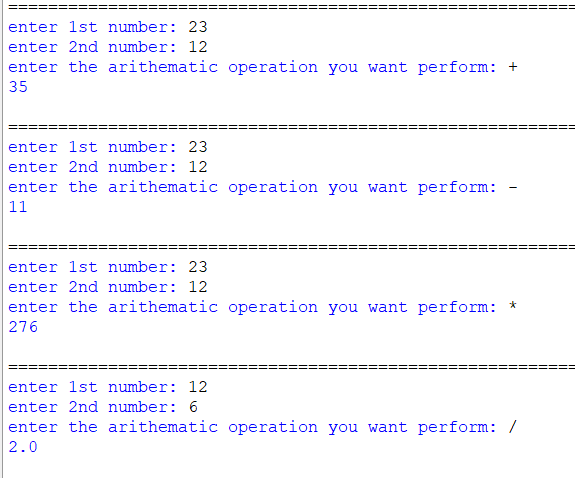


**9. write modular program to perform four basic operation on two numbers.**

CODE:

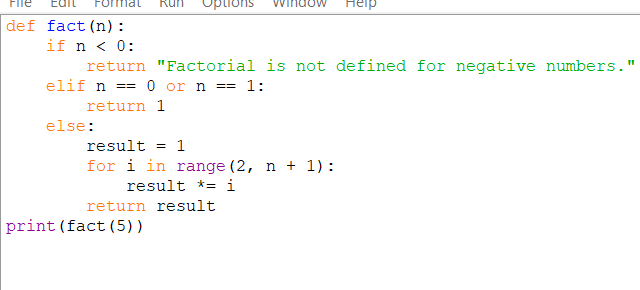


OUTPUT:

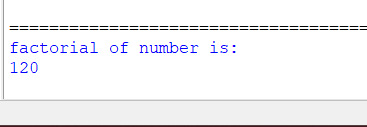


**10. Write a program to create function to calculate factorial of number.**

CODE:

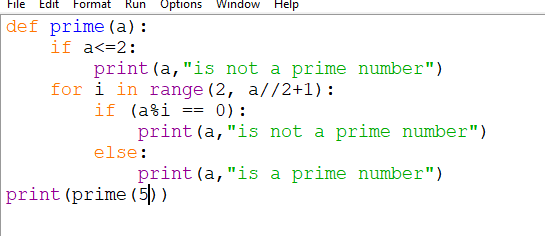


OUTPUT:

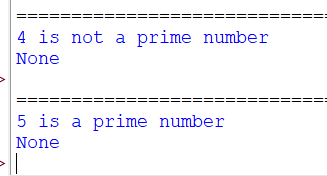


**11. Write a program to check entered number is prime or not**

CODE:

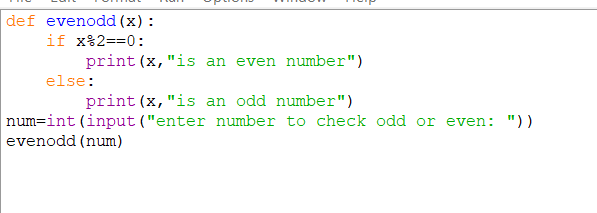


OUTPUT:

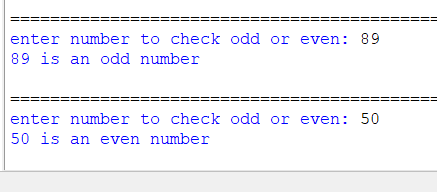


**12. Write a program to check number is even or odd**

CODE:

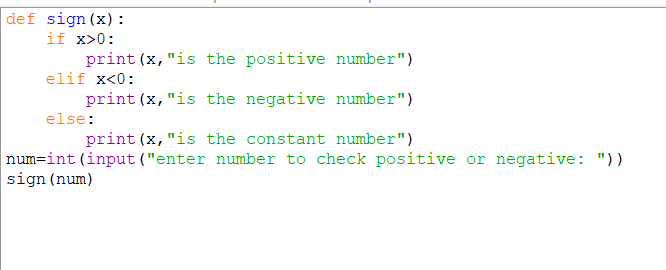


OUTPUT:

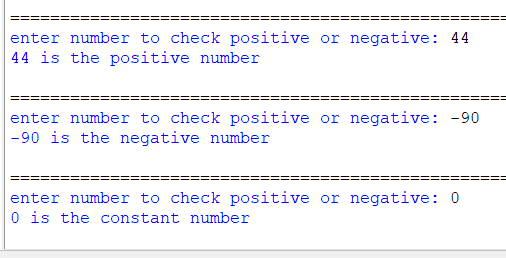


**13. Program to check number is positive or negative**

CODE:



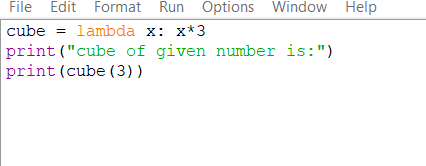
OUTPUT:



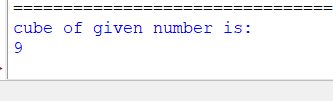
**ANONYMOUS FUNTIONS**

**1.Create anonymous function to print cube of a number**

CODE:



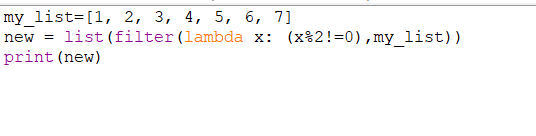
OUTPUT:



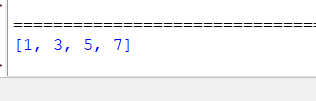
**2.Create a function to use filter function to segregate**

**odd numbers**

CODE:

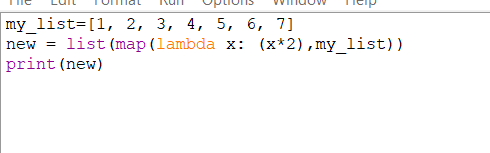


OUTPUT:

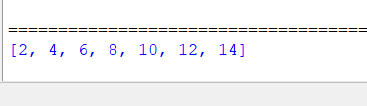


**3.Use map function to double element given in list**

CODE:

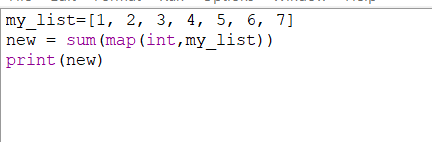


OUTPUT:

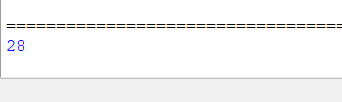


**4.Use map function to sum elements in given list**

CODE:

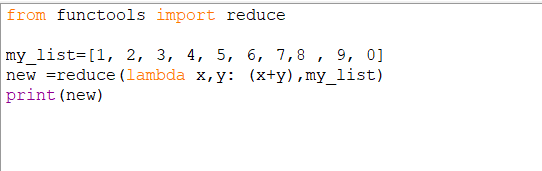


OUTPUT:

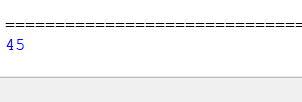


**5.Use reduce function to sum 10 numbers in list**

CODE:



OUTPUT:

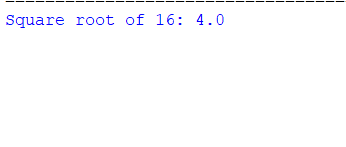
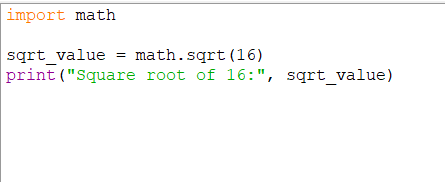


**MODULES**

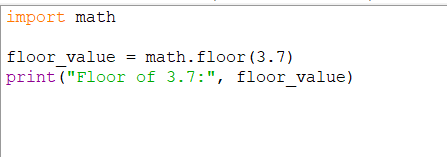
**1) USE FOLLOWING MODULES AND EXPLORE ANY FIVE METHODS RELATED TO IT.**

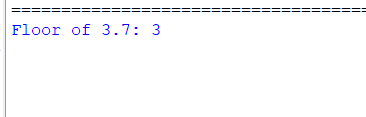
**○ MATH**

Math.sqrt:

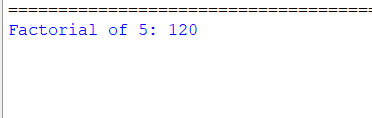
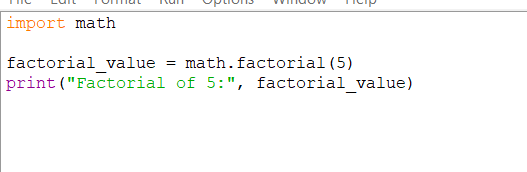


Math.floor:

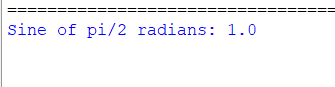
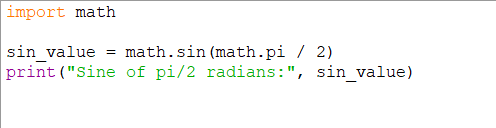




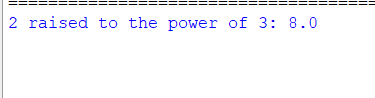
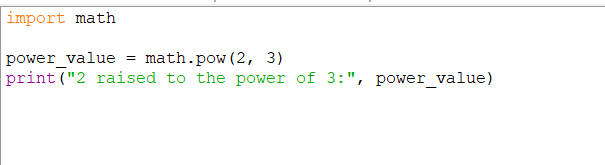
Math.factorial:



Math.sin:

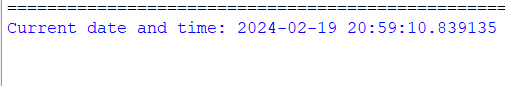
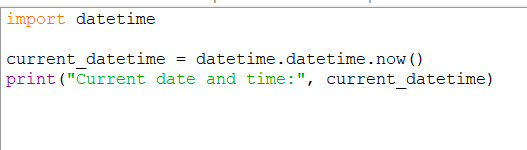


Math.pow:

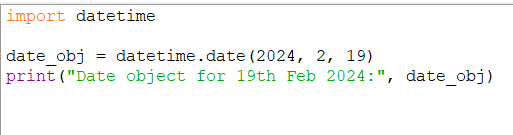


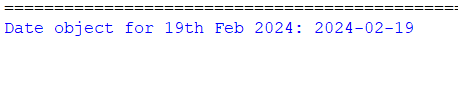
**○ DATETIME:**

Datetime.datetime.now:

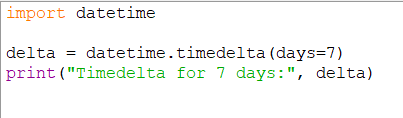


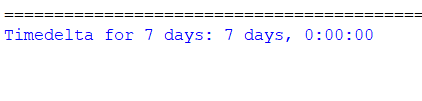
Datetime.date:



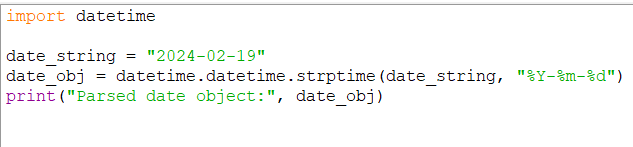


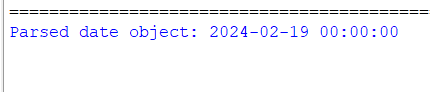
Datetime.timedelta:



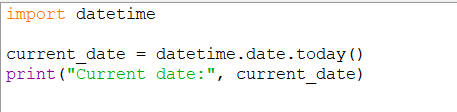


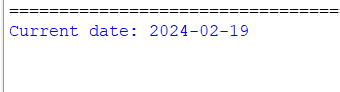
Datetime.strptime:





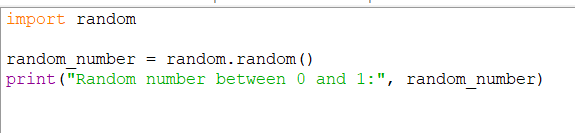
Datetime.today:

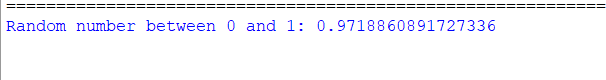




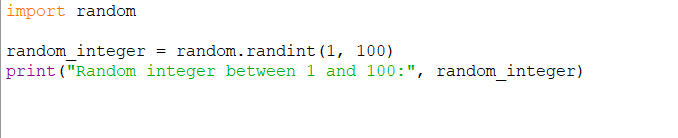
**○ RANDOM:**

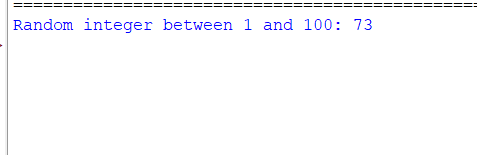
Random.random:



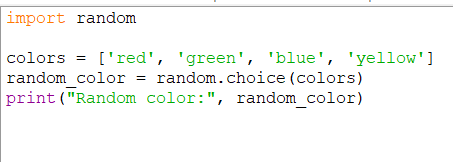


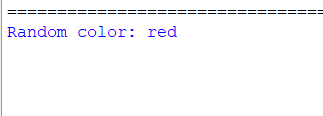
Random.randint:



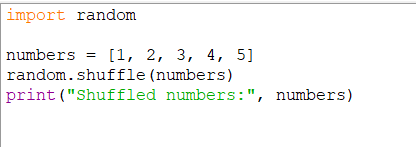


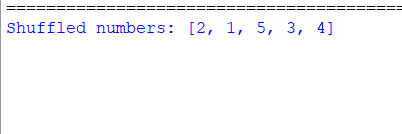
 random.choice:



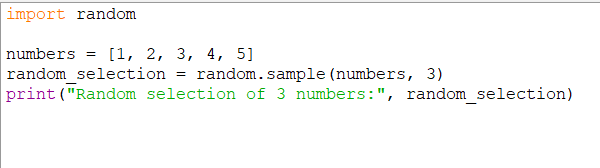


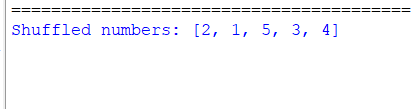
Random.shuffle:





Random.sample:





**○ CALENDAR**

**○ OS**

**○ JSON**

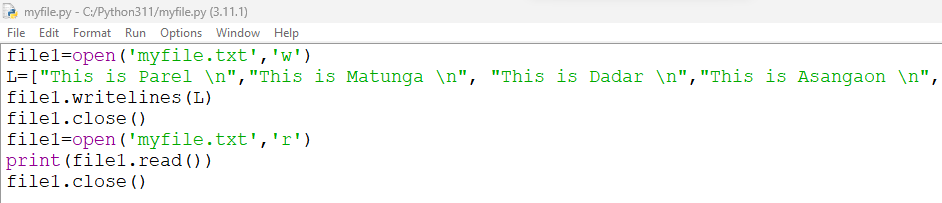
**○ MATPLOTLIB**

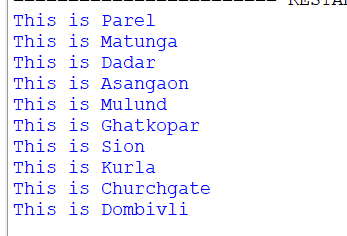
**○ SEABORN**

**○ PLOTLY**

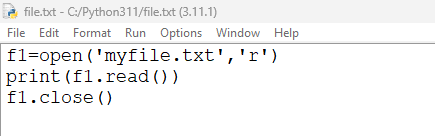
**PRACTICAL 5: FILE HANDLING**

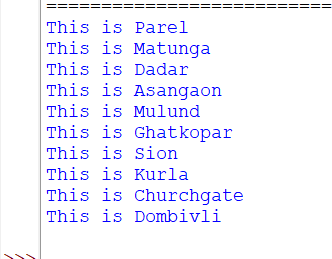
1. Write a program to create a file of ten lines of your choice ,every line should be on different line.



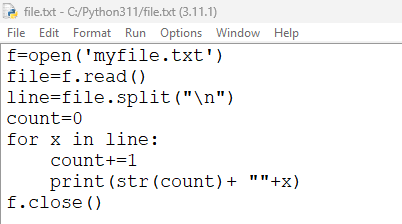


2. Write a program to read the entire contents of a file with read method.



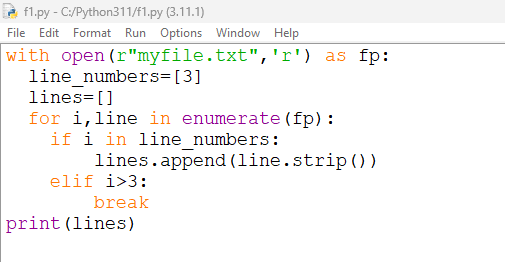


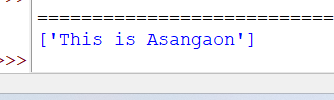
3. Write a program to display the lines along with the line numbers.



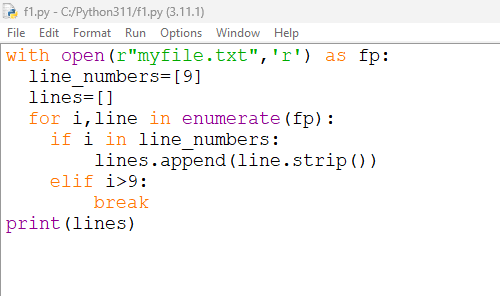


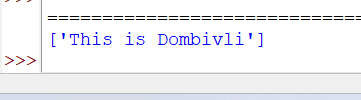
4. Write a program to display the fourth line.



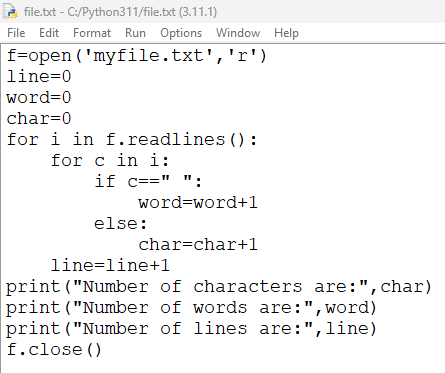


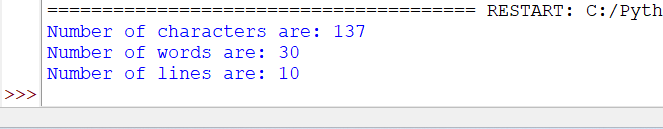
5. Write a program to display the last line.





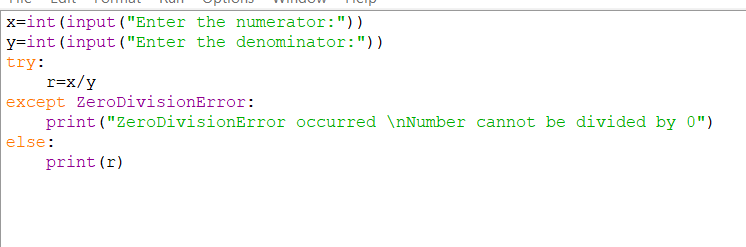
6. Write a program to count the number of lines, words, characters.

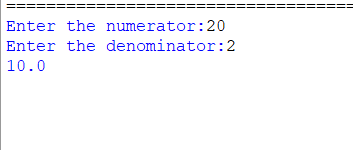


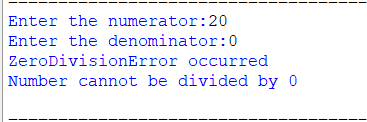


Practical 6: Exception Handling

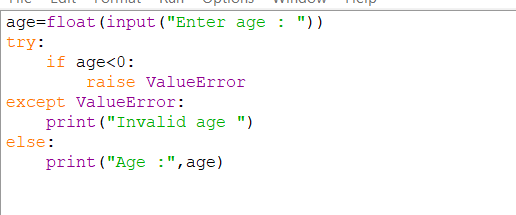
**) ZeroDivisionError**

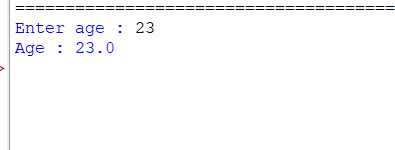


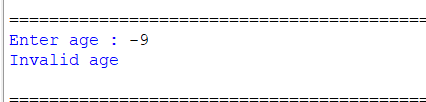




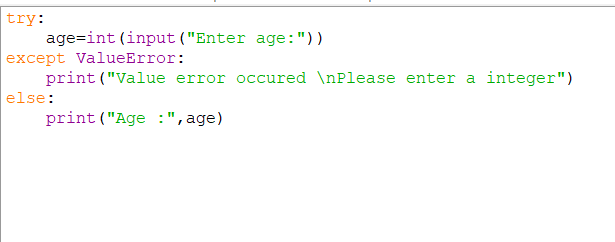
**2) Raise Value error**

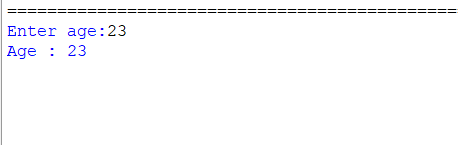
****

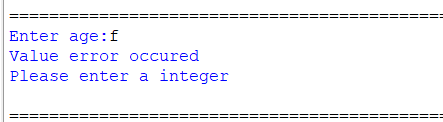
****

****

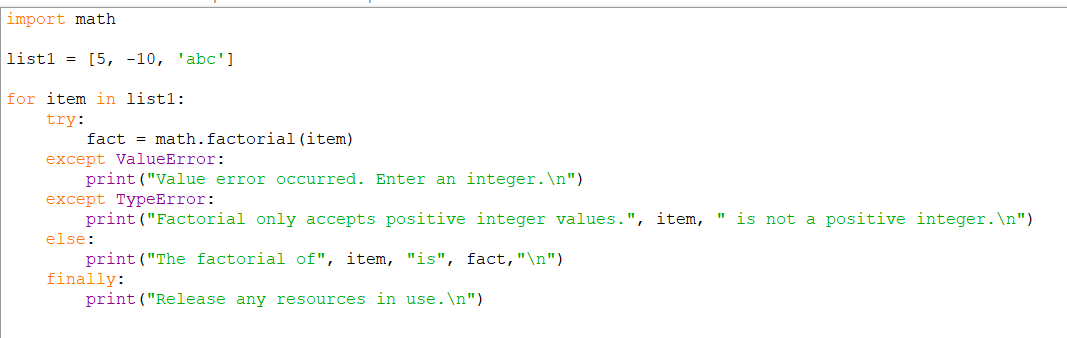
**3) Handling Value error**

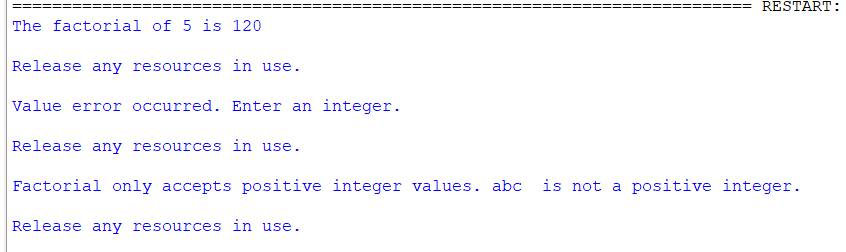
****

****

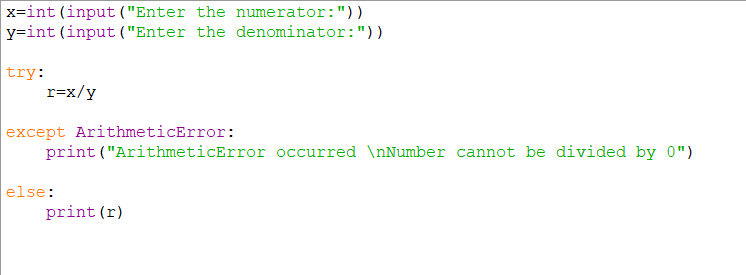
****

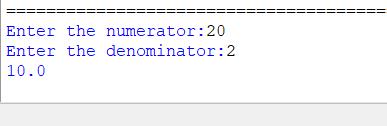
**4) Multiple exception**

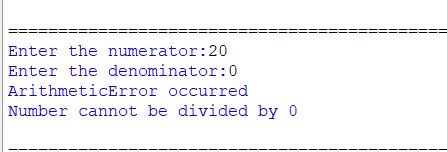
****

****

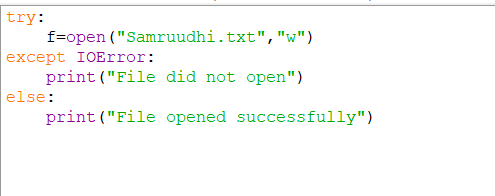
**5) Arithmetic error - base class of ZeroDivisionError**

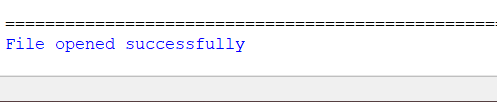
****

****

****

**6) IO error**

****

****

Practical 7: Regular Expression

1.Write a program to print the words in the string “My college name is Ruia” by naming the

groups.

import re

# Corrected string with five words

a = "My college name is Ruia College"

# Adjusted regular expression to match five words b =

re.search('(?P<first>.\*)\s(?P<second>.\*)\s(?P<third>.\*)\s(?P<fourt h>.\*)\s(?P<fifth>.\*)', a)

# Printing the matched groups

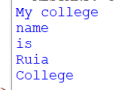
print(b.group('first'))

print(b.group('second'))

print(b.group('third'))

print(b.group('fourth'))

print(b.group('fifth'))

**

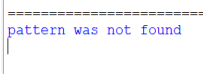
2.TO check pattern

import re

text="xyzabbpqv"

if re.match("ab+",text):

print("pattern was found") else:

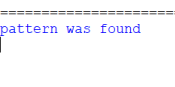
print("pattern was not found") **import re

text="xyzabbpqv"

if re.search("ab+",text):

print("pattern was found") else:

print("pattern was not found")

import re

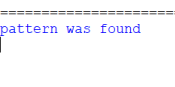
text="xyzabbpqv"

if re.findall("ab+",text):

print("pattern was found")

else:

print("pattern was not found")

**

3.Accept string from from user and check whether vowels or consonants

import re

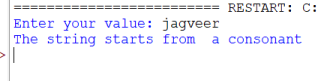
val = input("Enter your value: ")

if re.match('[aeiouAEIOU]',val):

print("The string starts from a vowel")

else:

print("The string starts from a consonant")

**

4) Consider the following string:“From: rasika.ruia@gmail.com for in 11-12-2015”

Display sender name, Display domain, Display the date of email.

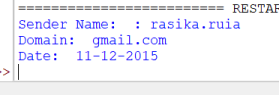
import re

a="From: rasika.ruia@gmail.com for in 11-12-2015" b=re.search("(?P<sender>:\s.\*)@(?P<domain>.\*)\s(?P<spc>.\*\s.\* )\s(?P<date>\d.\*)",a)

print("Sender Name: ",b.group("sender"))

print("Domain: ",b.group("domain"))

print("Date: ",b.group("date"))

**

5.Write a Python program to check that a string contains only a certain set of characters (in this case a-z, A

Z and 0-9).

import re

def specific\_char(string):

charRe = re.compile(r'[^a-zA-Z0-9]')

string = charRe.search(string)

return not bool(string)

print(specific\_char("ABCDEFabcdef123450"))

print(specific\_char("\*&%@#!}{"))

**

6.Write a Python program that matches a string that has an a followed by one or more b's.

import re

def text\_match(text):

patterns = 'ab+?'

if re.search(patterns, text):

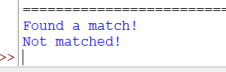
return 'Found a match!'

else:

return('Not matched!')

print(text\_match("ab"))

print(text\_match("acc"))

**

7.Write a Python program to find sequences of one upper case letter followed by lower case letters.

import re

def text\_match(text):

patterns = '[A-Z]+[a-z]+$'

if re.search(patterns, text):

return 'Found a match!'

else:

return('Not matched!')

print(text\_match("AaBbGg"))

print(text\_match("Python"))

print(text\_match("python"))

print(text\_match("PYTHON"))

print(text\_match("aA"))

print(text\_match("Aa"))

8.Write a Python program where a string will start with a specific number.

import re

def match\_num(string):

text = re.compile(r"^5")

if text.match(string):

return True

else:

return False

print(match\_num('5-2345861'))

print(match\_num('6-2345861'))

**

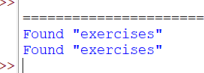
9.Write a Python program to find the substrings within a string.

import re

text = 'Python exercises, PHP project, C# exercises' pattern = 'exercises'

for match in re.findall(pattern, text):

print('Found "%s"' % match)

**

10.Write a Python program that matches a word containing 'z'.

import re

def text\_match(text):

patterns = '\w\*z.\w\*'

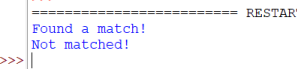
if re.search(patterns, text):

return 'Found a match!'

else:

return('Not matched!')

print(text\_match("The quick brown fox jumps over the lazy dog.")) print(text\_match("Python Exercises."))

**

11.Write a Python program to match if two words from a list of words starting with letter 'P'.

import re

# Sample strings.

words = ["Python PHP", "Java JavaScript", "c c++"]

for w in words:

m = re.match("(P\w+)\W(P\w+)", w)

# Check for success

if m:

print(m.groups())

*https://lh7-us.googleusercontent.com/U5680CfNbkitBk4dD6RProG5WnFx3dtRP3nWbHpou6gF2FPBRZp5rPp3x2GiAZWLuGuhDghSd9ltqXdUzuDG93JFbxVLVsJVZ5bpibEZnccGhVMtkRsy3o7ZmMV62uksHuHtO6r0WWeWGGLKer7kjYU*

12.Write a Python program to find all words starting with 'a' or 'e' in a given string.

import re

# Input.

text = "The following example creates an ArrayList with a capacity of 50 elements. Four elements are then added to the ArrayList and the ArrayList is trimmed accordingly."

#find all the words starting with 'a' or 'e'

list = re.findall("[ae]\w+", text)

# Print result.

print(list)

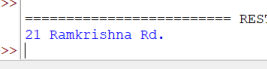
*https://lh7-us.googleusercontent.com/BE-pn66rRn_XLwvxu4gRzsZfK4u3ItVUH8lsZhWb7qG9DGRFbYl0nufHar4oL9V24P31od7fR5Go_OcLTR_JngmaLjY5nGk6D-zzNWjsOeRRq_QQ8uRek-JKRtYB_FNaQ3D5jb2QLVcWr-s6RmJNGyg*

13.Write a Python program to abbreviate 'Road' as 'Rd.' in a given string.

import re

street = '21 Ramkrishna Road'

print(re.sub('Road$', 'Rd.', street))



14.Write a Python program to replace all occurrences of space, comma, or dot with a colon.

import re

text = 'Python Exercises, PHP exercises.'

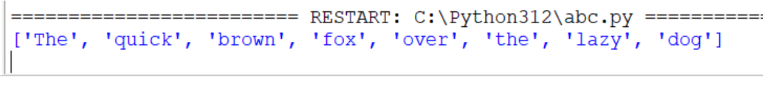
print(re.sub("[ ,.]", ":", text))

*https://lh7-us.googleusercontent.com/q1vrdl8Zs3EMy_ywPvWjC1H0ObF54uwdZv6HX0tStexJJGRkKbXOonr0lEkAIf23V51jXokaKbC95ha3NrPDMfCNaXh20HixAGsnxGaAX-ntuKy7xLRM4mWfKbHQ8XwrvmO1i9TcYHZpBBDN33XnGq4*

15.Write a Python program to find all three, four, five characters long words in a string.

import re

text = 'The quick brown fox jumping over the lazy dog.' print(re.findall(r"\b\w{3,5}\b", text))



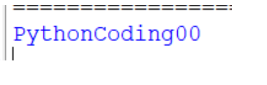
16.Write a Python program to remove everything except alphanumeric characters from a string.

import re

text1 = '\*\*//Python Exercises// - 00. '

pattern = re.compile('[\W\_]+')

print(pattern.sub('', text1))



17.Write a program to find words starting with ‘d’ or ‘p’ in a line

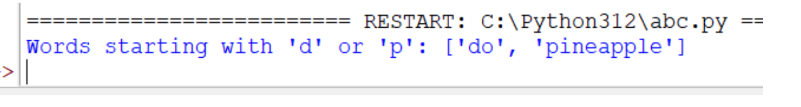
import re

line = "do you know if he wants pineapple juice"

pattern = r"\b[dp]\w\*"

matches = re.findall(pattern, line)

print("Words starting with 'd' or 'p':", matches)



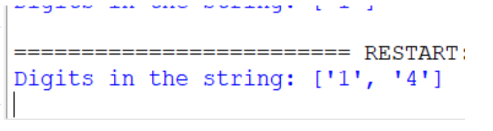
18.Write a program to print digits from a string “one 1 two 2 3 three”

import re

string = "hello fish one 1 4"

digits = re.findall(r'\d+', string)

print("Digits in the string:", digits)



19.From a string “voorheesville” find out words starting with ‘vi’

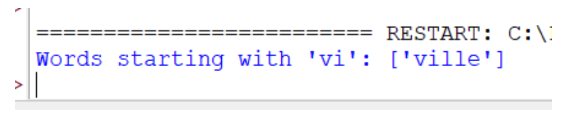
import re

string = "voorheesville"

pattern = r"\bvi\w\*"

matches = re.findall(pattern, string)

print("Words starting with 'vi':", matches)



20.Write a program to demonstrate match function which will have a list=["dog dot","dodon't","dumb-dumb","no match"]

import re

words = ["dog dot", "dodon't", "dumb-dumb", "no match"] pattern = r"\b\w+\b"

for word in words:

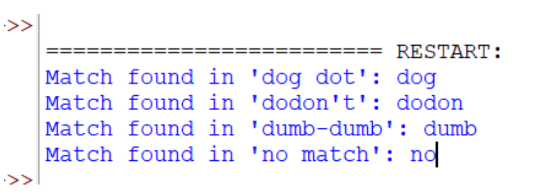
match = re.match(pattern, word)

if match:

print(f"Match found in '{word}': {match.group()}")

else:

print(f"No match found in '{word}'")



21.Consider a string “I like mango”. Find whether string ends with ‘o’ or not and extract ‘like’ from string

import re

string = "I like mango"

# Check if the string ends with 'o'

if re.search(r'o\Z', string):

print("The string ends with 'o'")

else:

print("The string does not end with 'o'")

# Extract 'like' from the string

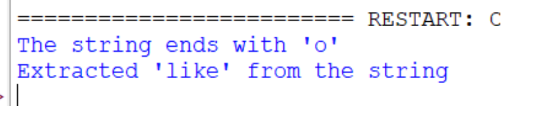
match = re.search(r'\blike\b', string)

if match:

print("Extracted 'like' from the string")

else:

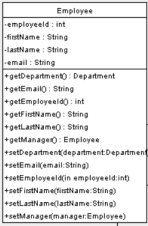
print("'like' not found in the string")



Practical 8: Oops Concept

1.Write a python program to demonstrate encapsulation concept 1.1 Without constructor

1.2 With Constructor



Code:

1.1 Without Constructor

class Employee:

\_employeeId=0

\_firstName=""

\_lastName=""

\_email=""

\_department=""

def getDepartment(self):

print("Department Name:",self.\_department)

def getEmail(self):

print("Email Id:",self.\_email)

def getEmployeeId(self):

print("Employee Id:",self.\_employeeId)

def getName(self):

print("Name of Employee

is:",self.\_firstName,"",self.\_lastName)

def setDepartment(self,depName):

self.\_department=depName

def setEmail(self,email):

self.\_email=email

def setEmployeeId(self,empId):

self.\_employeeId=empId

def setName(self,fname,lname):

self.\_firstName=fname

self.\_lastName=lname

def getDetails(self):

print("Id:",self.\_employeeId,"\nName:",self.\_firstName,"",self.\_la stName,"\nEmail:",self.\_email,"\nDepartment:",self.\_departmen t)

emp1=Employee()

emp1.setName("Shruti","Singh")

emp1.setEmployeeId(1)

emp1.setDepartment("IT")

emp1.setEmail("shanti@gmail.com")

emp1.getDetails()

Output:

1.2With Constructor

Code:

class Employee:

def

\_\_init\_\_(self,employeeId,firstName,lastName,email,department): self.\_employeeId=employeeId #protected variables self.\_firstName=firstName

self.\_lastName=lastName

self.\_email=email

self.\_\_department=department #private Variables def getDepartment(self):

print("Department Name:",self.\_department)

def getEmail(self):

print("Email Id:",self.\_email)

def getEmployeeId(self):

print("Employee Id:",self.\_employeeId)

def getName(self):

print("Name of Employee

is:",self.\_firstName,"",self.\_lastName)

def setDepartment(self,depName):

self.\_department=depName

def setEmail(self,email):

self.\_email=email

def setEmployeeId(self,empId):

self.\_employeeId=empId

def setName(self,fname,lname):

self.\_firstName=fname

self.\_lastName=lname

def getDetails(self):

print("Id:",self.\_employeeId,"\nName:",self.\_firstName,"",self.\_la stName,"\nEmail:",self.\_email,"\nDepartment:",self.\_departmen t)

emp1=Employee(1,"Shruti","Singh","shanti@gmail.com","IT") emp1.getDetails()

emp1.getName()

emp2=Employee(2,"","","","")

emp2.setName("Amit","Singh")

emp2.setDepartment("Sales")

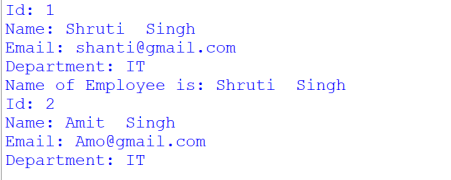
emp2.setEmail("Amo@gmail.com")

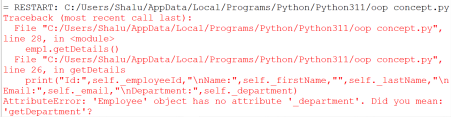
emp2.setEmployeeId(2)

emp2.\_department="IT"

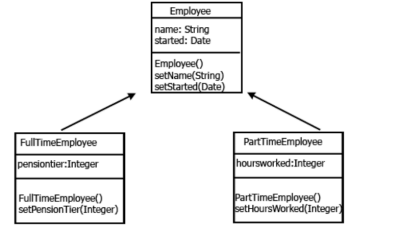
emp2.getDetails()

Output:





2.Write a program to Demonstrate Inheritance concept 2.1 Single Inheritance

Code:

class Employee:

def \_\_init\_\_(self, name, date):

self.name = name

self.joinDate = date

def setName(self, name):

self.name = name

def setJoinDate(self, date):

self.joinDate = date

def show(self):

print("Name:", self.name, "Joining Date:", self.joinDate)

class FullTimeEmployee(Employee):

def \_\_init\_\_(self, name, date, pensionTier):

Employee.\_\_init\_\_(self, name, date)

self.pensionTier = pensionTier

def getAmount(self):

print("Pension Amount:", self.pensionTier)

class PartTimeEmployee(Employee):

def \_\_init\_\_(self, name, date, hoursWorked):

Employee.\_\_init\_\_(self, name, date)

self.hoursWorked = hoursWorked

def setHoursWorked(self, amount):

self.hoursWorked = amount

def getAmount(self):

print("Hourly Amount:", self.hoursWorked)

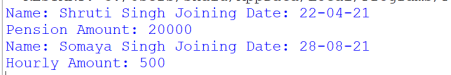
obj1 = FullTimeEmployee("Shruti Singh", "22-04-21", 20000) obj1.show()

obj1.getAmount()

obj2 = PartTimeEmployee("Somaya Singh", "28-08-21", 500) obj2.show()

obj2.getAmount()

Output:



2.Code:

class Teacher:

def \_\_init\_\_(self,subName,tName):

self.subName = subName

self.tName = tName

def show(self):

print("Subject Name:",self.subName,"\nTeacher Name:",self.tName)

class VisitingTeacher(Teacher):

def \_\_init\_\_(self,subName,tName,hrsRate):

Teacher.\_\_init\_\_(self,subName,tName)

self.hrsRate = hrsRate

def displayDetails(self):

Teacher.show(self)

print("Hours Rate:",self.hrsRate)

class ContractTeacher(Teacher):

def \_\_init\_\_(self,subName,tName,expDate):

super(). \_\_init\_\_(subName,tName)

self.expDate = expDate

def displayDetails(self):

Teacher.show(self)

print("Contract Expire Date:",self.expDate)

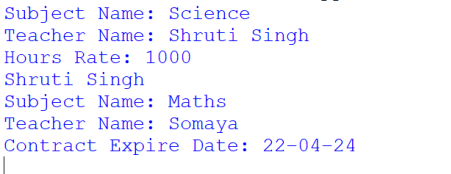
obj1 = VisitingTeacher("Science","Shruti Singh",1000) obj1.displayDetails()

print(obj1.tName)

obj2 = ContractTeacher("Maths","Somaya","22–04–24") obj2.show()

print("Contract Expire Date:",obj2.expDate)

Output:

2.2.Multiple Inheritance

Code:

class MotherProperty:

def \_\_init\_\_(self,mLocation,mValue):

self.mLocation = mLocation

self.mValue = mValue

def displayMotherProperty(self):

print("Mother property at ",self.mLocation,"and its value is ", self.mValue)

class FatherProperty:

def \_\_init\_\_(self,fLocation,fValue):

self.fLocation = fLocation

self.fValue = fValue

def displayMotherProperty(self):

print("Father property at ",self.fLocation,"and its value is ", self.fValue)

class ChildProperty(MotherProperty,FatherProperty): def

\_\_init\_\_(self,mLocation,mValue,fLocation,fValue,cLocation,cValu e):

super().\_\_init\_\_(mLocation,mValue)

FatherProperty.\_\_init\_\_(self,fLocation,fValue)

self.cLoation = cLocation

self.cValue = cValue

def displayTotalValue(self):

print("Super excited to know property

value:",self.cValue+self.mValue+self.fValue)

obj1 = ChildProperty("juhu",9400000,"Ghatkopar",19500000,"Malabar Hill",7400000)

obj1.displayMotherProperty()

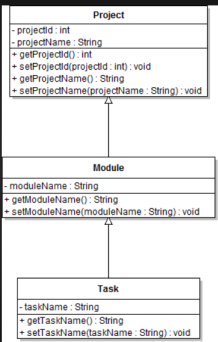
obj1.displayTotalValue()

Output:

https://lh7-us.googleusercontent.com/eh-uFqWkj3T-eNv1u70yHu1-M8jKdceEHHcbocm7DPIEIGku4qoUOLyUOt6nka9fhDgy-SDs7rW05-XHRm4f_niiOHayiZsnET9qzFSGSXWL6FRF_s1udkxYVw9eXTmFdq8Sg6pRik-xfnFMSNfGs7o

2.3

Multilevel Inheritance



Code:

class Project:

def \_\_init\_\_(self,projectId,projectName):

self.projectId = projectId

self.projectName = projectName

def getProjectId(self):

print("Project Id:",self.projectId)

def setProjectId(self,pId):

self.projectId = pId

def getProjectName(self):

print("Project Name:",self.projectName)

def setProjectName(self,pName):

self.projectName = pName

class Module(Project):

def \_\_init\_\_(self,projectId,projectName,moduleName): Project.\_\_init\_\_(self,projectId,projectName)

self.moduleName = moduleName

def getModuleName(self):

print("Module Name:",self.moduleName)

def setModuleName(self,mName):

self.moduleName = mName

class Task(Module):

def

\_\_init\_\_(self,projectId,projectName,moduleName,taskName): Module.\_\_init\_\_(self,projectId,projectName,moduleName)

self.taskName = taskName

def getTaskName(self):

print("Task Name:",self.taskName)

def setTaskName(self,tName):

self.taskName = tName

obj1 = Task("p1","Inventory Management System","sql","Database Connection")

obj1.getProjectName()

obj1.getModuleName()

obj1.getTaskName()

Output:

https://lh7-us.googleusercontent.com/aHITv0_Qd2mNyKe320V310S4lZOEFHrAlMr6jvyAxQaTERe9X0fr8zGU2IMerk8_5tszyaMXw4aFDVibRT_H0VfJGEV_ULUZH5yn1I363W75achF_YVy3yfbvs78mBsA29Ms6jpiOmnv1UAMZZzZ9oY

3.Write a program to show Polymorphism

Code:

#Method Overriding

from math import pi

class Shape:

def \_\_init\_\_(self, name):

self.name = name

def area(self):

pass

def fact(self):

return "I am a two-dimensional shape."

def \_\_str\_\_(self):

return self.name

class Square(Shape):

def \_\_init\_\_(self, length):

super().\_\_init\_\_("Square")

self.length = length

def area(self):

return self.length\*\*2

def fact(self):

return "Squares have each angle equal to 90 degrees."

class Circle(Shape):

def \_\_init\_\_(self, radius):

super().\_\_init\_\_("Circle")

self.radius = radius

def area(self):

return pi\*self.radius\*\*2

a = Square(4)

b = Circle(7)

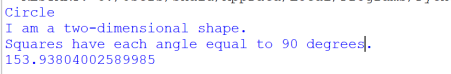
print(b)

print(b.fact())

print(a.fact())

print(b.area())

Output:



4.Write a Python program to create a class representing a Circle. Include methods to calculate its area and perimeter. Code:

import math

class Circle:

def \_\_init\_\_(self,radius):

self.r = radius

def area(self):

return math.pi\*self.r\*self.r

def perimeter(self):

return 2\*math.pi\*self.r

obj1 = Circle(2.5)

print("Area of circle is:",obj1.area())

print("Circumference of circle is:",obj1.perimeter()) Output:

https://lh7-us.googleusercontent.com/Zbp_hMJ6ZtOm-7xiULcr0aGQvVXYvempjuvZwVTQQDGEG3uZX1GKpeFleijsgn1T3ivZWPk63hpR7HmdGSQTVP7kNfAdkfaTOseNkg9NxOCI4htOvzMSkr4wNwOjLOcu3G380f2k9XzazmAyMIiyWks

5.Write a Python program to create a calculator class. Include methods for basic arithmetic operations. Code:

class Calculator:

def \_\_init\_\_(self,num1,num2):

self.num1 = num1

self.num2 = num2

def add(self):

return self.num1+self.num2

def sub(self):

return self.num1-self.num2

def product(self):

return self.num1\*self.num2

def Division(self):

return self.num1/self.num2

obj1 =Calculator(18,8)

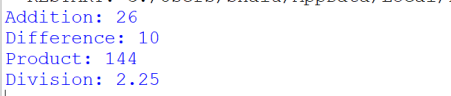
print("Addition:",obj1.add())

print("Difference:",obj1.sub())

print("Product:",obj1.product())

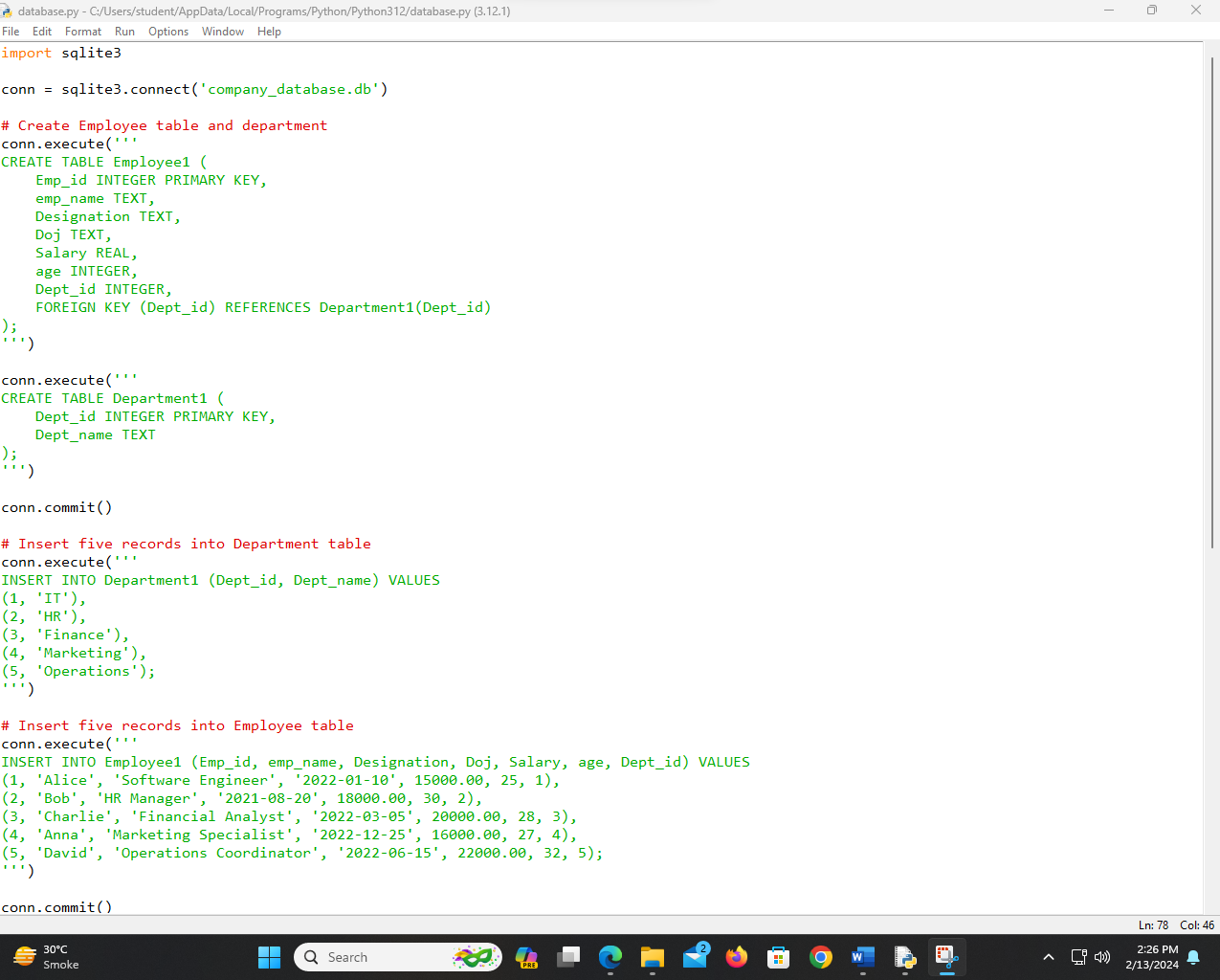
print("Division:",obj1.Division())

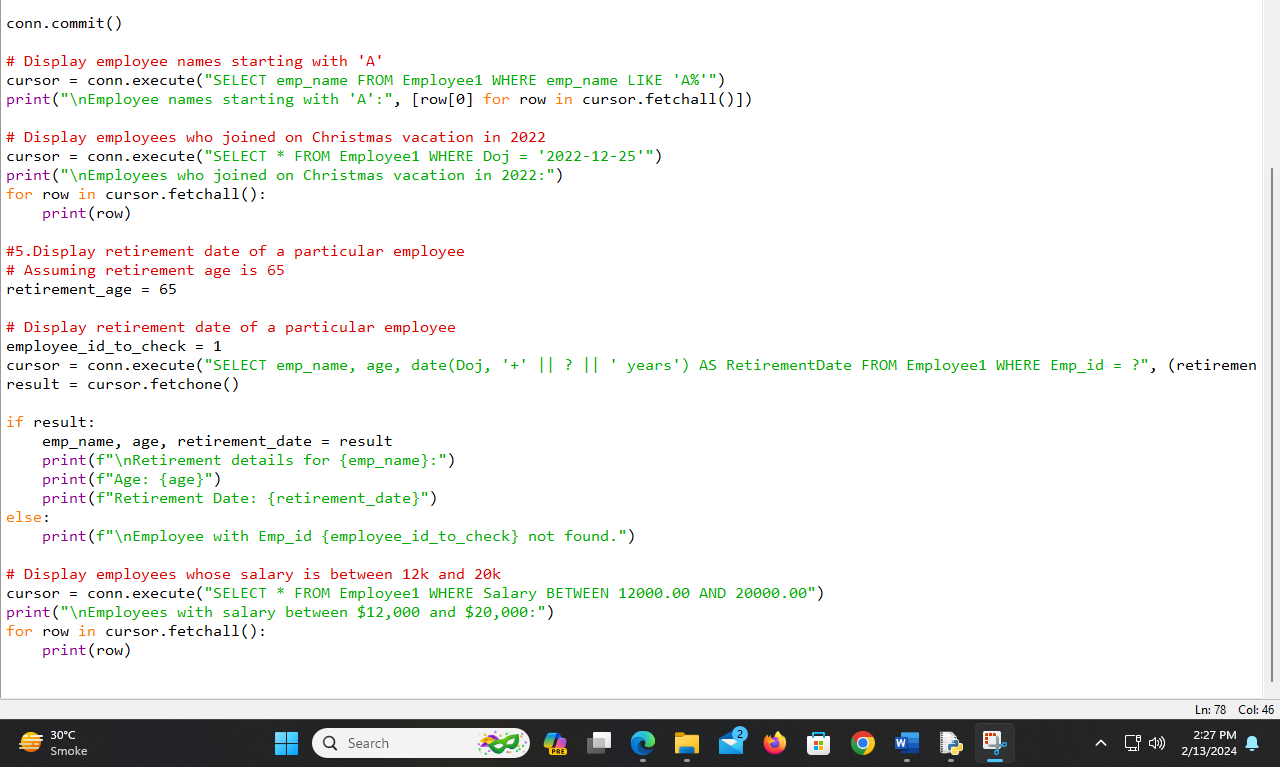
Output:

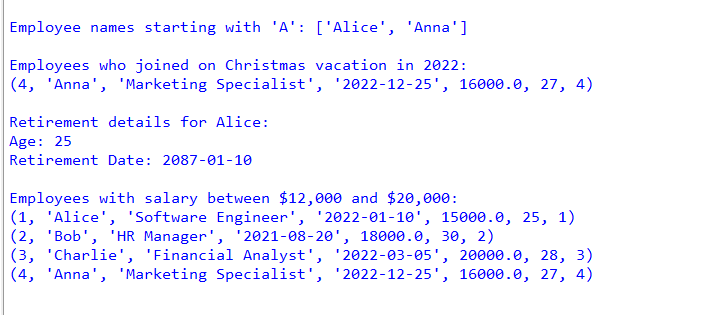


**Practical 9:Database connectivity**

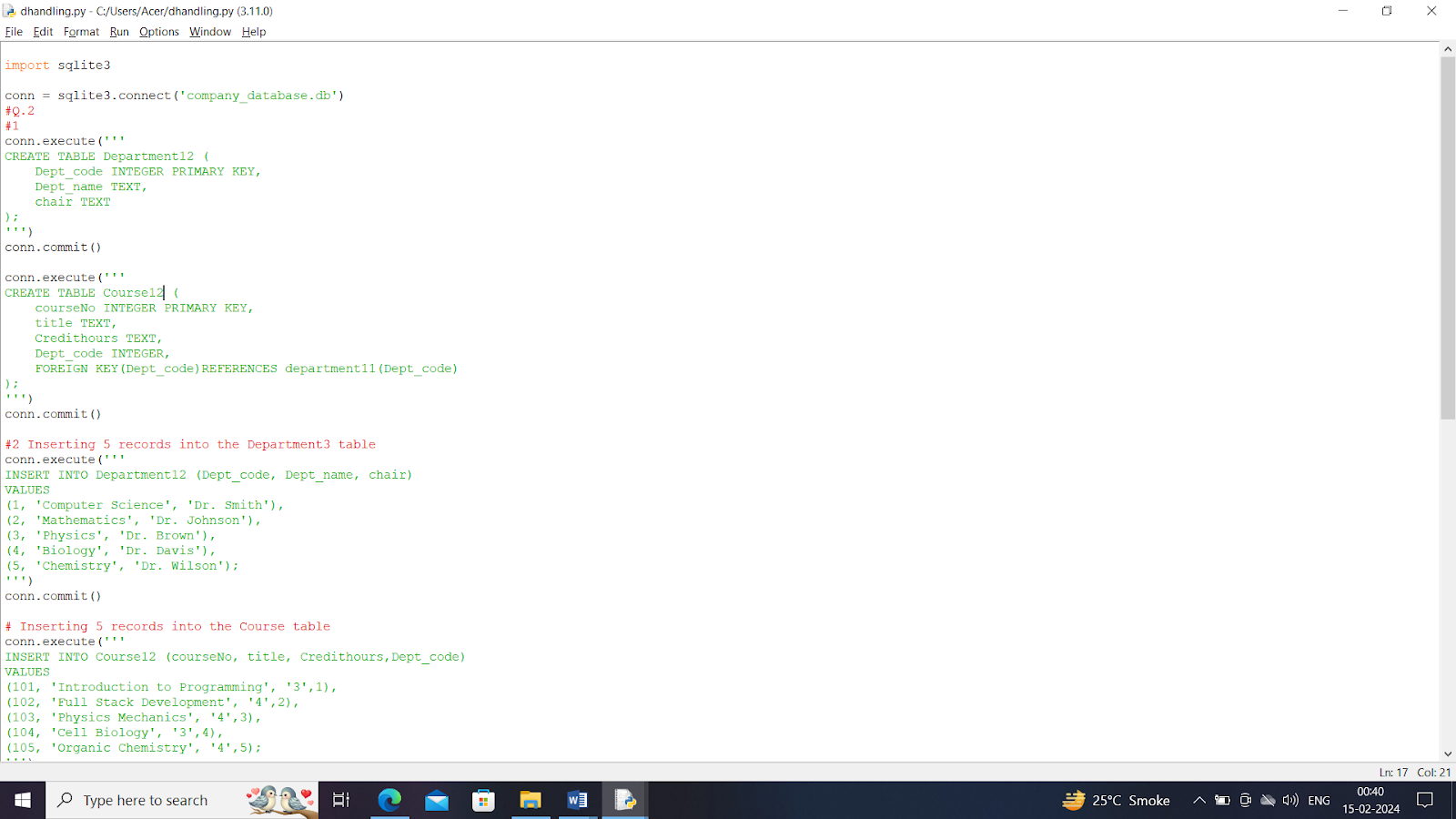
1. Create the following table (Department, Employee).
2. Create above tables in sqlite.
3. Insert five records in each table.
4. Display emp names whose names start with ‘A’.
5. Display employee who joins in Christmas vacation in 2022.
6. Display retirement date of a particular employee.
7. Display emp those salary is between 12k and 20k.

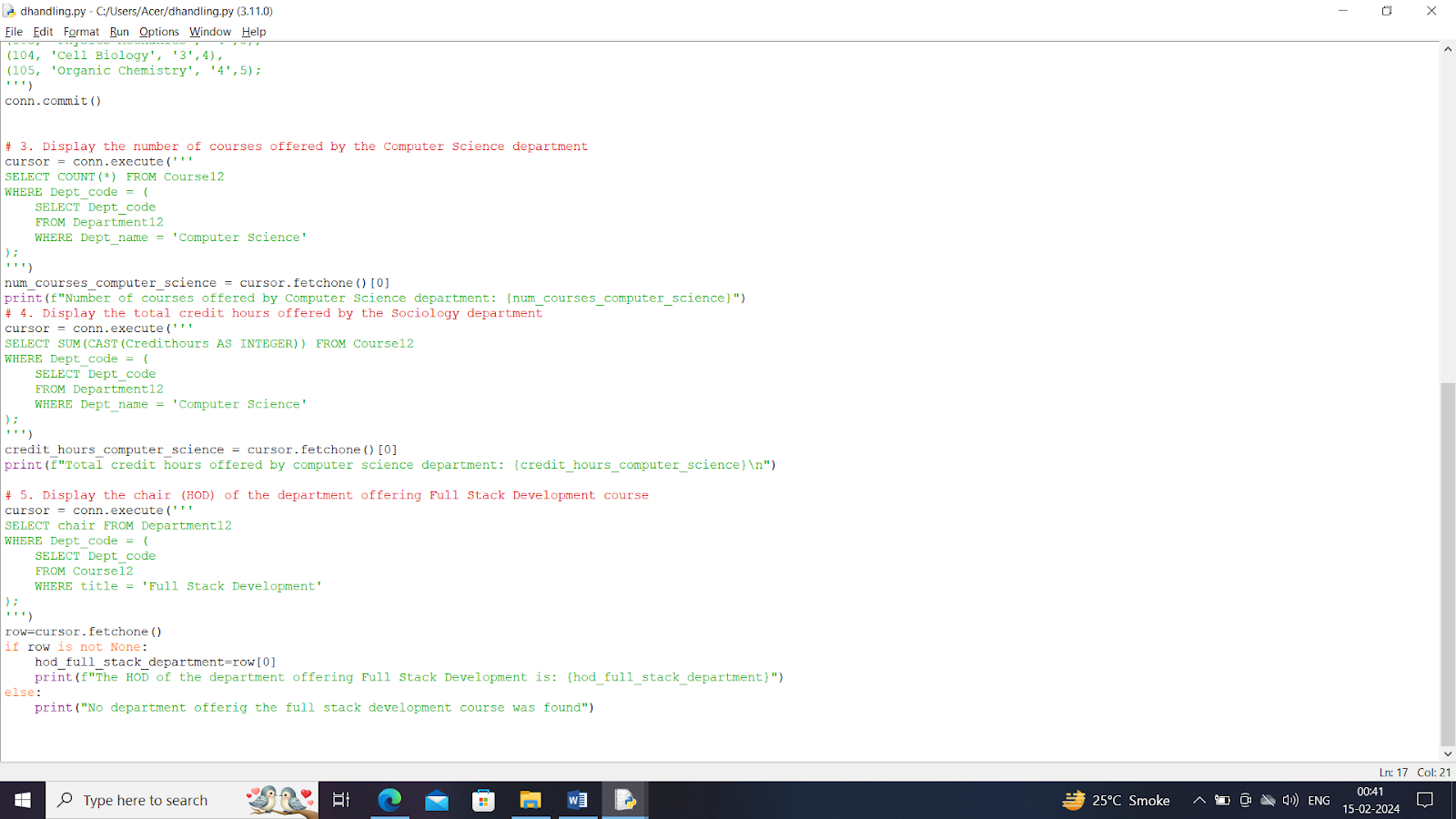


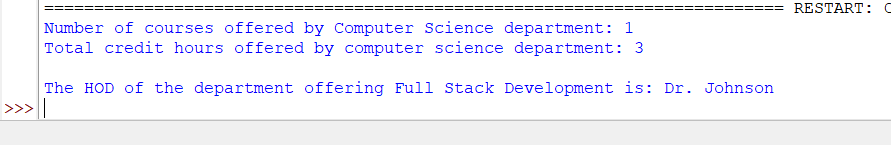




1. Create table according to ER  diagram
2. Create above tables.
3. Insert 5 records
4. Display number of courses offered by computer science department.
5. Display credit hours offered by sociology department.
6. Display who is the HOD of department who is offering full stack development course.







**Practical 10: GUI Programming**

1. Design a login page

import tkinter as tk

from tkinter import messagebox

def submit():

username = entry\_username.get()

password = entry\_password.get()

if username == "aditya" and password == "12345":

messagebox.showinfo("Login Successful", "Welcome, " + username + "!")

else:

messagebox.showerror("invalid","wrong username and password")

def cancel():

entry\_username.delete(0, tk.END)

entry\_password.delete(0, tk.END)

root = tk.Tk()

root.title("Login Page")

label\_username = tk.Label(root, text="Username:")

label\_username.grid(row=0, column=0, padx=10, pady=10)

entry\_username = tk.Entry(root)

entry\_username.grid(row=0, column=1, padx=10, pady=10)

label\_password = tk.Label(root, text="Password:")

label\_password.grid(row=1, column=0, padx=10, pady=10)

entry\_password = tk.Entry(root, show="\*")

entry\_password.grid(row=1, column=1, padx=10, pady=10)

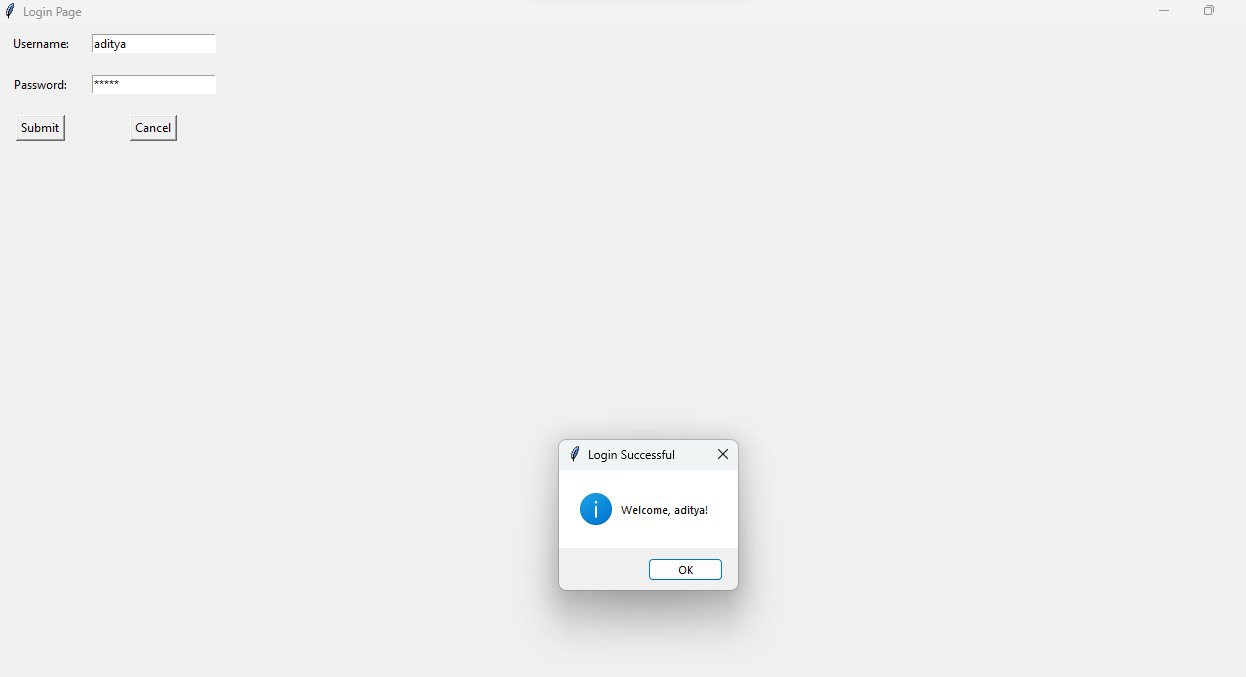
button\_submit = tk.Button(root, text="Submit", command=submit)

button\_submit.grid(row=2, column=0, padx=10, pady=10)

button\_cancel = tk.Button(root, text="Cancel", command=cancel)

button\_cancel.grid(row=2, column=1, padx=10, pady=10)

root.mainloop()



2.design biodata

import tkinter as tk

from tkinter import messagebox

def submit():

name = entry\_name.get()

address = entry\_address.get("1.0", tk.END)

dob = entry\_dob.get()

gender = gender\_var.get()

qualification = entry\_qualification.get()

hobbies = entry\_hobbies.get("1.0", tk.END)

messagebox.showinfo("Submitted Data", f"Name: {name}\nAddress: {address}\nDate of Birth: {dob}\nGender: {gender}\nQualification: {qualification}\nHobbies: {hobbies}")

def cancel():

entry\_name.delete(0, tk.END)

entry\_address.delete("1.0", tk.END)

entry\_dob.delete(0, tk.END)

gender\_var.set("") # Reset gender selection

entry\_qualification.delete(0, tk.END)

entry\_hobbies.delete("1.0", tk.END)

root = tk.Tk()

root.title("Biodata Form")

label\_name = tk.Label(root, text="Name:")

label\_name.grid(row=0, column=0, padx=10, pady=5, sticky="w")

entry\_name = tk.Entry(root)

entry\_name.grid(row=0, column=1, padx=10, pady=5)

label\_address = tk.Label(root, text="Address:")

label\_address.grid(row=1, column=0, padx=10, pady=5, sticky="w")

entry\_address = tk.Text(root, height=4, width=30)

entry\_address.grid(row=1, column=1, padx=10, pady=5)

label\_dob = tk.Label(root, text="Date of Birth:")

label\_dob.grid(row=2, column=0, padx=10, pady=5, sticky="w")

entry\_dob = tk.Entry(root)

entry\_dob.grid(row=2, column=1, padx=10, pady=5)

label\_gender = tk.Label(root, text="Gender:")

label\_gender.grid(row=3, column=0, padx=10, pady=5, sticky="w")

gender\_var = tk.StringVar()

gender\_var.set("")

male\_radio = tk.Radiobutton(root, text="Male", variable=gender\_var, value="Male")

male\_radio.grid(row=3, column=1, padx=5, pady=5, sticky="w")

female\_radio = tk.Radiobutton(root, text="Female", variable=gender\_var, value="Female")

female\_radio.grid(row=3, column=1, padx=5, pady=5, sticky="e")

other\_radio = tk.Radiobutton(root, text="other", variable=gender\_var, value="other")

other\_radio.grid(row=3, column=2, padx=5, pady=5, sticky="s")

label\_qualification = tk.Label(root, text="Qualification:")

label\_qualification.grid(row=4, column=0, padx=10, pady=5, sticky="w")

entry\_qualification = tk.Entry(root)

entry\_qualification.grid(row=4, column=1, padx=10, pady=5)

label\_hobbies = tk.Label(root, text="Hobbies:")

label\_hobbies.grid(row=5, column=0, padx=10, pady=5, sticky="w")

entry\_hobbies = tk.Text(root, height=4, width=30)

entry\_hobbies.grid(row=5, column=1, padx=10, pady=5)

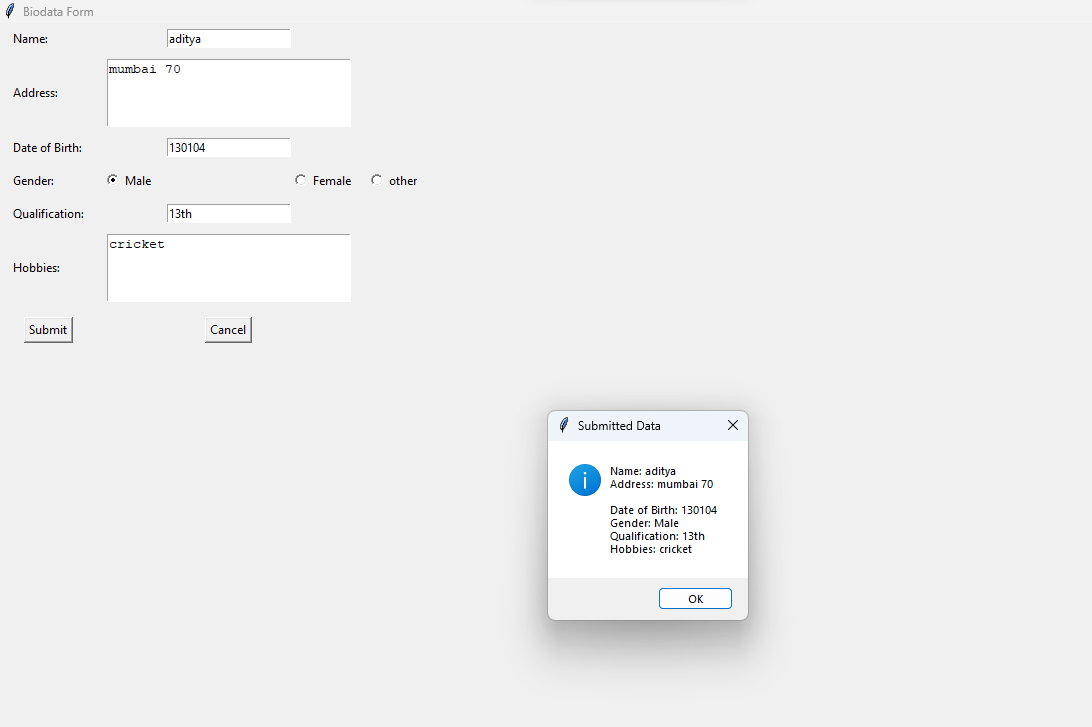
button\_submit = tk.Button(root, text="Submit", command=submit)

button\_submit.grid(row=6, column=0, padx=10, pady=10)

button\_cancel = tk.Button(root, text="Cancel", command=cancel)

button\_cancel.grid(row=6, column=1, padx=10, pady=10)

root.mainloop()



1. Write a program on artmatic opration on gui

import tkinter as tk

from tkinter import messagebox

def perform\_operation():

num1 = float(entry\_num1.get())

num2 = float(entry\_num2.get())

operation = operation\_var.get()

if operation == "+":

result = num1 + num2

elif operation == "-":

result = num1 - num2

elif operation == "\*":

result = num1 \* num2

elif operation == "/":

if num2 == 0:

messagebox.showerror("Error", "Division by zero is not allowed!")

return

else:

result = num1 / num2

messagebox.showinfo("Result", f"Result of {num1} {operation} {num2} = {result}")

root = tk.Tk()

root.title("Basic Arithmetic Operations")

label\_num1 = tk.Label(root, text="Number 1:")

label\_num1.grid(row=0, column=0, padx=10, pady=5)

entry\_num1 = tk.Entry(root)

entry\_num1.grid(row=0, column=1, padx=10, pady=5)

label\_num2 = tk.Label(root, text="Number 2:")

label\_num2.grid(row=1, column=0, padx=10, pady=5)

entry\_num2 = tk.Entry(root)

entry\_num2.grid(row=1, column=1, padx=10, pady=5)

label\_operation = tk.Label(root, text="Operation:")

label\_operation.grid(row=2, column=0, padx=10, pady=5)

operation\_var = tk.StringVar()

operation\_var.set("+") # Default operation is addition

operations = ["+", "-", "\*", "/"]

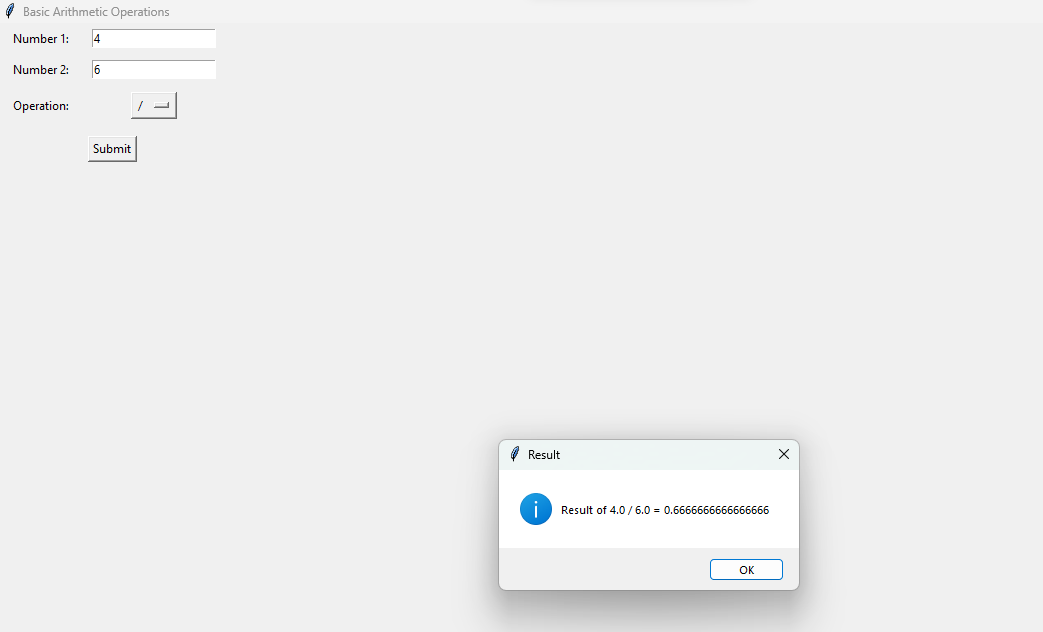
operation\_menu = tk.OptionMenu(root, operation\_var, \*operations)

operation\_menu.grid(row=2, column=1, padx=10, pady=5)

button\_submit = tk.Button(root, text="Submit", command=perform\_operation)

button\_submit.grid(row=3, column=0, columnspan=2, padx=10, pady=10)

root.mainloop()



4.write a program to display any five shapes

import tkinter as tk

root = tk.Tk()

root.title("Shapes")

canvas = tk.Canvas(root, width=400, height=400)

canvas.pack()

canvas.create\_rectangle(50, 50, 150, 100, fill="red")

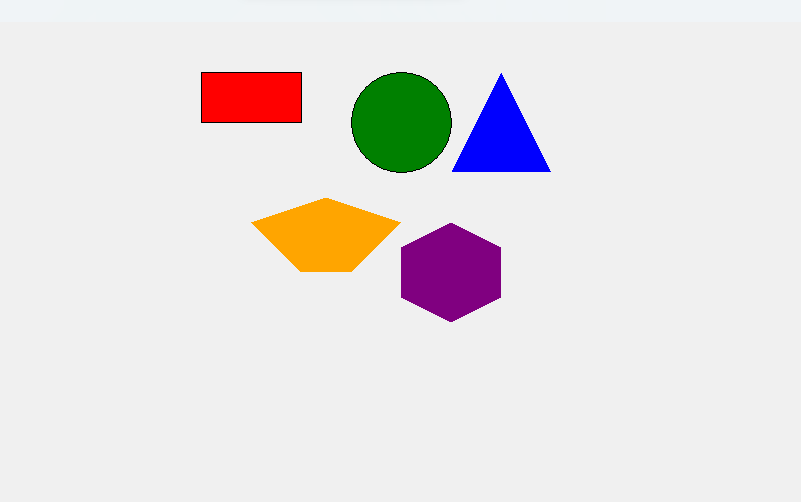
canvas.create\_oval(200, 50, 300, 150, fill="green")

canvas.create\_polygon(350, 50, 300, 150, 400, 150, fill="blue")

canvas.create\_polygon(100, 200, 150, 250, 200, 250, 250, 200, 175, 175, fill="orange")

canvas.create\_polygon(300, 200, 350, 225, 350, 275, 300, 300, 250, 275, 250, 225, fill="purple")

root.mainloop()



5.wirte a program to create gui of notepad editor of window.

import tkinter as tk

from tkinter import scrolledtext

from tkinter import filedialog

from tkinter import messagebox

def new\_file():

if len(text\_area.get("1.0", tk.END)) > 0:

if messagebox.askyesno("Save File", "Do you want to save changes?"):

save\_file()

text\_area.delete("1.0", tk.END)

def open\_file():

file\_path = filedialog.askopenfilename()

if file\_path:

with open(file\_path, "r") as file:

text = file.read()

text\_area.delete("1.0", tk.END)

text\_area.insert("1.0", text)

def save\_file():

file\_path = filedialog.asksaveasfilename(defaultextension=".txt")

if file\_path:

with open(file\_path, "w") as file:

text = text\_area.get("1.0", tk.END)

file.write(text)

def about():

messagebox.showinfo("About", "Simple Notepad made with Tkinter")

root = tk.Tk()

root.title("Notepad")

text\_area = scrolledtext.ScrolledText(root, wrap=tk.WORD)

text\_area.pack(expand=True, fill="both")

menu\_bar = tk.Menu(root)

root.config(menu=menu\_bar)

file\_menu = tk.Menu(menu\_bar, tearoff=0)

menu\_bar.add\_cascade(label="File", menu=file\_menu)

file\_menu.add\_command(label="New", command=new\_file)

file\_menu.add\_command(label="Open...", command=open\_file)

file\_menu.add\_command(label="Save", command=save\_file)

file\_menu.add\_separator()

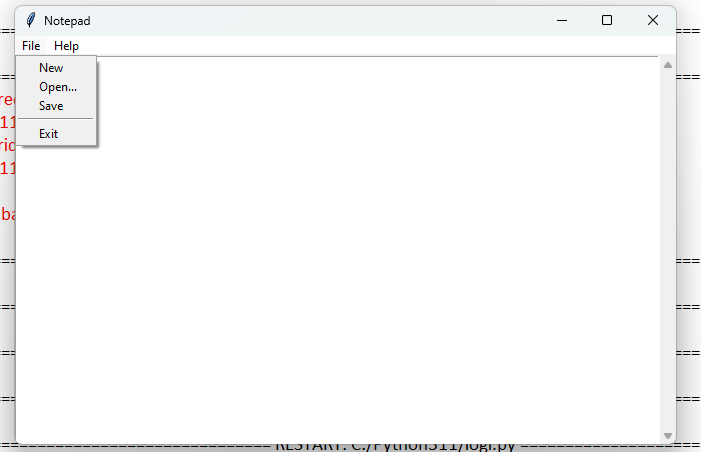
file\_menu.add\_command(label="Exit", command=root.quit)

help\_menu = tk.Menu(menu\_bar, tearoff=0)

menu\_bar.add\_cascade(label="Help", menu=help\_menu)

help\_menu.add\_command(label="About", command=about)

root.mainloop()



6.wirte a program accept temperature in celcius and convert into Fahrenheit

import tkinter as tk

def convert\_to\_fahrenheit():

celsius = float(entry\_celsius.get())

fahrenheit = (celsius \* 9/5) + 32

label\_result.config(text=f"{celsius}°C is {fahrenheit:.2f}°F")

root = tk.Tk()

root.title("Temperature Converter")

label\_celsius = tk.Label(root, text="Enter temperature in Celsius:")

label\_celsius.pack(pady=5)

entry\_celsius = tk.Entry(root)

entry\_celsius.pack(pady=5)

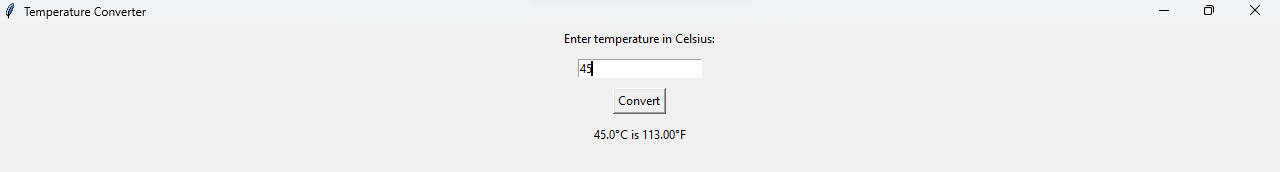
button\_convert = tk.Button(root, text="Convert", command=convert\_to\_fahrenheit)

button\_convert.pack(pady=5)

label\_result = tk.Label(root, text="")

label\_result.pack(pady=5)

root.mainloop()



7.write a program to create a natural seanary

import tkinter as tk

def draw\_scenery():

canvas.create\_rectangle(0, 0, 600, 400, fill="sky blue", outline="")

canvas.create\_oval(500, 50, 550, 100, fill="yellow", outline="")

canvas.create\_rectangle(0, 300, 600, 400, fill="green", outline="")

canvas.create\_rectangle(0, 200, 600, 300, fill="blue", outline="")

canvas.create\_polygon(200, 300, 300, 200, 400, 300, fill="blue", outline="")

root = tk.Tk()

root.title("Natural Scenery")

canvas = tk.Canvas(root, width=600, height=400)

canvas.pack()

draw\_scenery()

root.mainloop()



8.write a program to display cartoon images using tkinter

import tkinter as tk

def draw\_cartoon():

canvas.create\_oval(50, 50, 150, 150, fill="peach puff", outline="black")

canvas.create\_oval(80, 80, 100, 100, fill="white", outline="black")

canvas.create\_oval(120, 80, 140, 100, fill="white", outline="black")

canvas.create\_oval(85, 85, 95, 95, fill="black")

canvas.create\_oval(125, 85, 135, 95, fill="black")

canvas.create\_polygon(110, 100, 115, 130, 105, 130, fill="peach puff", outline="black")

canvas.create\_arc(90, 110, 130, 130, start=0, extent=-180, fill="peach puff", outline="black")

canvas.create\_rectangle(70, 150, 130, 250, fill="blue", outline="black")

canvas.create\_line(70, 180, 30, 220, width=5)

canvas.create\_line(130, 180, 170, 220, width=5)

canvas.create\_line(90, 250, 70, 300, width=5)

canvas.create\_line(110, 250, 130, 300, width=5)

root = tk.Tk()

root.title("Cartoon Image")

canvas = tk.Canvas(root, width=200, height=350)

canvas.pack()

draw\_cartoon()

root.mainloop()

