**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Practical Name** | **Teacher Sign** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |
| 6 |  |  |
| 7 |  |  |

**Program 1:** **a) Write a Python Program to Calculate the Area of a Triangle**

**b) Write a Python Program to Swap Two Variables**

**c) Write a Python Program to Convert Celsius to Fahrenheit**

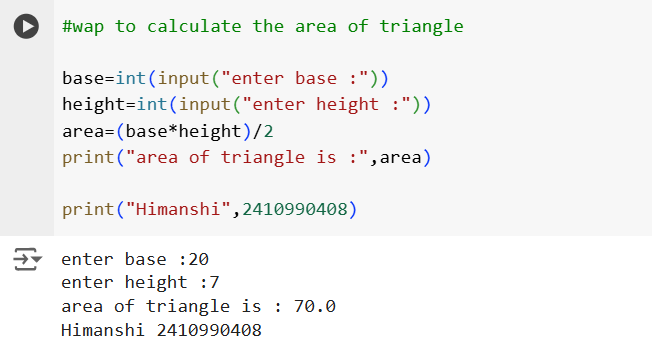
**Solution:**

Objective a) - to calculate the area of a triangle from the input values put in by the user

base=int(input("enter base: “)) #taking input from user height=int(input("enter height: “))

area=0.5\*base\*height #using formula 1/2\*b\*h print(area)

**Output-**

****

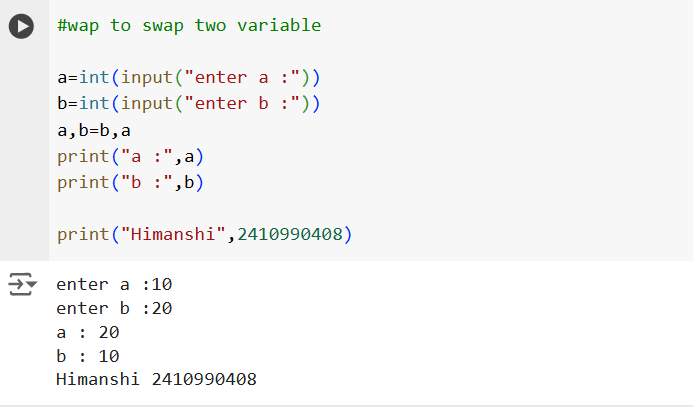
Objective b) - to swap the values of two variables assigned before and interchanging them with each other

a=10

b=20

a,b=b,a #swapping the value assigned to variable before print(a,b)

**Output-**

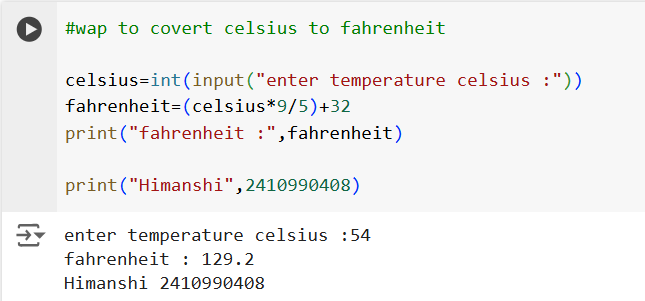


Objective c) - to covert temperature from celsius to fahrenheit from input given by user

celsius=int(input("enter temperature in celsius: “)) # taking input from user

fahrenheit=(celsius\*1.8)+32 # conversion from c to f print(fahrenheit)

**Output-**



**Program 2 : a.) Write a Python Program to Check if a Number is Odd or Even**

**b.) Write a Python Program to Check if a Number is Positive, Negative**

**or 0**

**c.) Write a Python Program to Check Armstrong Number**

**solution:**

Objective a) - to check the given input number by user is even or odd using conditional if else statement

n=int(input("enter a number: “)) # taking input from user

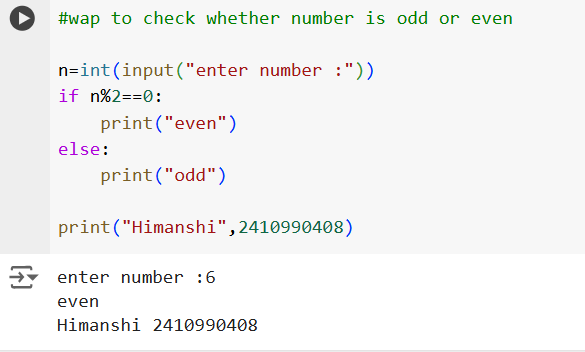
if n%2==0: # checking the number by dividing it by 2

print("even")

else:

print(“odd")

**Output-**



Objective b) - to check if a number is positive, negative or 0 taking input from user using conditional if else statement

n=int(input("enter a number: “)) # taking input from user

if n>0: # checking of the number

print("positive")

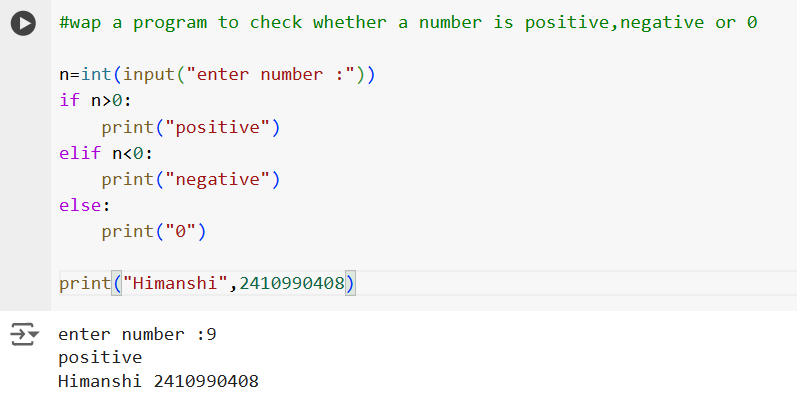
elif n<0:

print("negative")

else:

print(“zero")

**Output-**

****

Objective c) - to check whether number is an Armstrong number or not by taking input given by user

n=int(input("enter a number: “)) # taking input from user sum=0 # initialize sum temp=n # find sum of cube of each digit

while temp>0:

digit=temp//=10

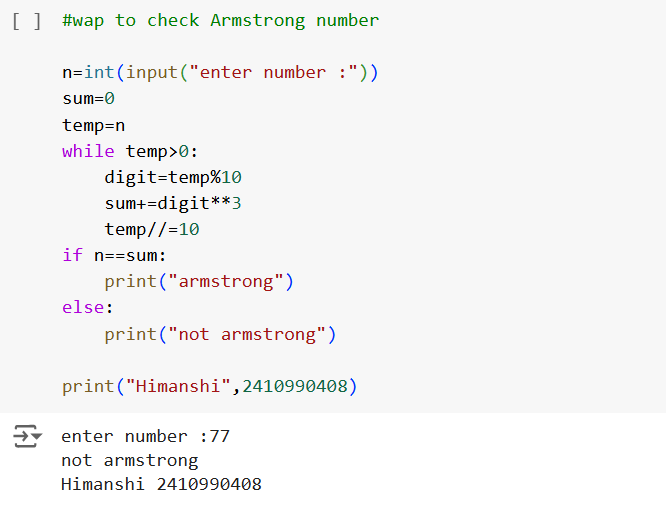
sum+=digit\*\*3

temp//=10 if n==sum: # display the result

print("armstrong number") else:

print("not an armstrong number”)

**Output-**

****

**Program 3 : a.) Write a Python program to check if a given number is Fibonacci number?**

**b.) Write a Python program to print cube sum of first n natural numbers.**

**c.) Write a Python program to print all odd numbers in a range.**

Solution :

Objective a) - to check the number is Fibonacci by taking input from user using function

def f(n): # defining a function n

if n<=0:

return False

a,b=0,1 # initialzing first two numbers 0 and 1 while a<=n: # checking the fibonacci number

if a==n:

return True

a,b=b,a+b

return False

n=int(input("enter a number: “)) # taking input from user

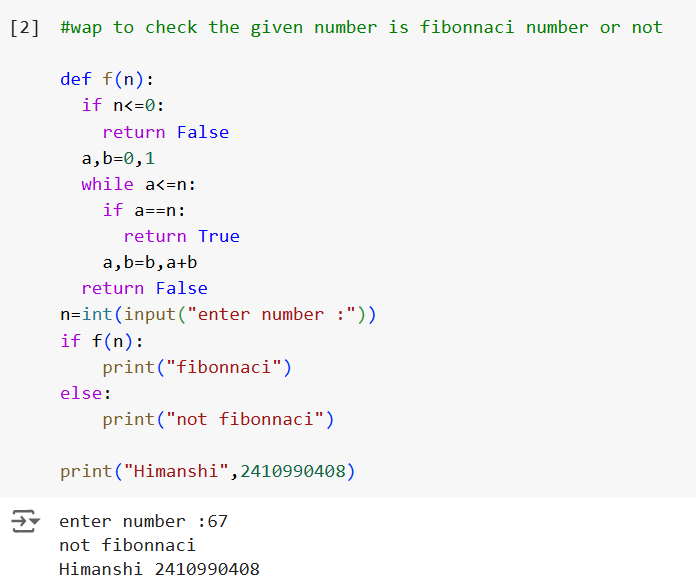
if f(n):

print("fibonnaci number")

else:

print("not a fibonnaci number”) # display of result

Output-

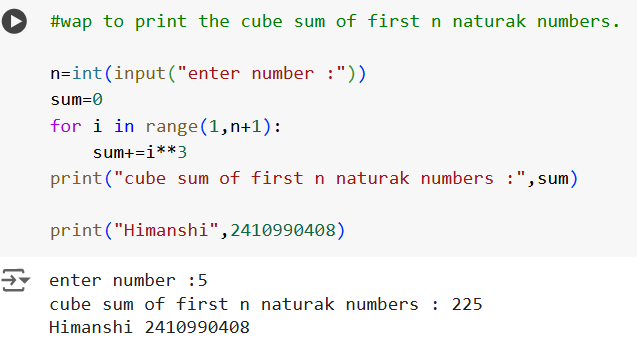


Objective b) - to print cube sum of first n natural numbers by taking input from user

n=int(input("enter a number: “)) # taking input from user sum=0 # intialzing sum = 0 for i in range(1,n+1):

sum+=i\*\*3 # adding the cube of number print(sum)

Output-



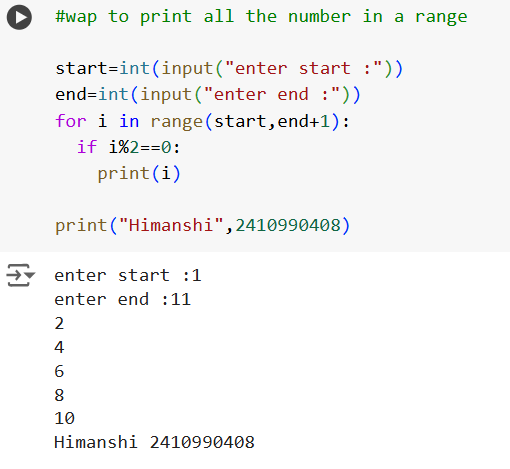
Objective c) - to print all odd numbers in a

start=int(input("enter start number: “)) # taking input from user end=int(input("enter end number: ")) for i in range(start,end+1):

if i%2!=0: #checking number divided by 2 not equal to zero

print(i)

Output-



**Program 4:**

**a.) Write a Python Program to Print Pascal Triangle**

**Hint: Enter number of rows: 4**

**1**

**1 1**

**1 2 1**

**1 3 3 1**

**b.) WAP to Draw the following Pattern for n number:**

**1 1 1 1 1**

**2 2 2 2**

**3 3 3**

**4 4**

**5**

Solution :

Objective a) - to Print Pascal Triangle by taking input from user

from math import factorial

n=int(input("enter a number: “)) # taking input from user

for i in range(n):

for j in range(n-i+1):

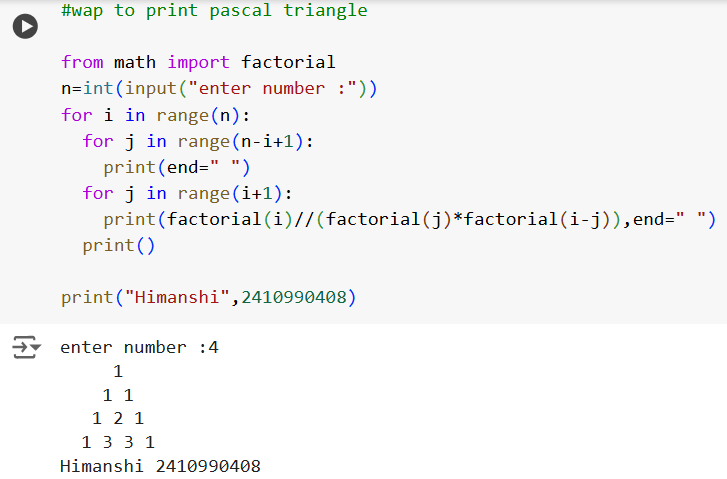
print(end=" “) # for left spacing

for j in range(i+1):

print(factorial (i)//(factorial(j)\*factorial(i-j)),end=" “) # nCr = n!/((n-r)!\*r!)

print() # for new line

Output-



*Objective b) -* to Draw the following Pattern for n number:

1. 1 1 1 1
2. 2 2 2
3. 3 3
4. 4

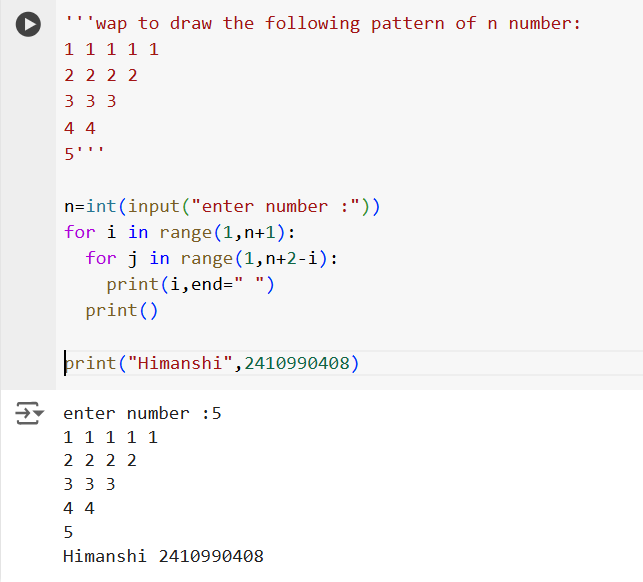
5

n=int(input("enter a number: “)) # taking input from user for i in range(1,n+1): for j in range(1,n+2-i):

print(i,end=" ")

print() # for new line

output-



**Program 5 : Write a program with a function that accepts a string from keyboard and create a new string after converting character of each word capitalized. For instance, if the sentence is “stop and smell the roses” the output should be “Stop And Smell The Roses**”

Solution :

### Objective - to covert the first character of each word capitalized of the string input by the user

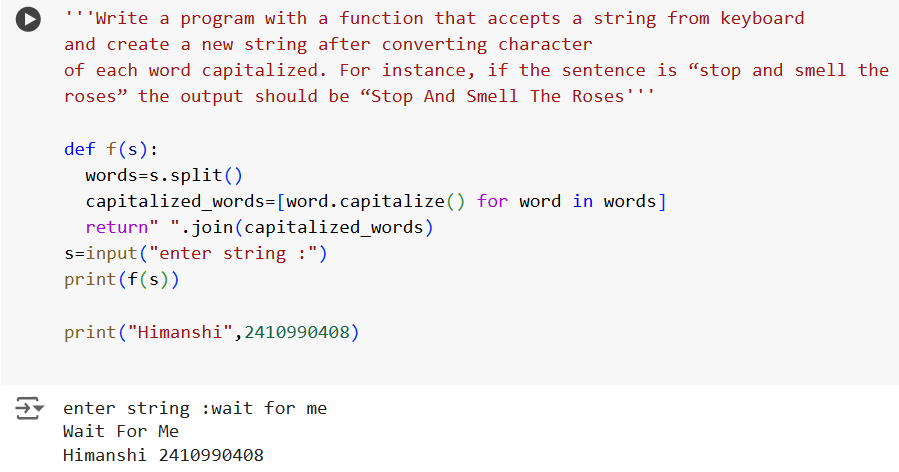
def f(s):

words=s.split()

capitalized\_words=[word.capitalize() for word in words]

return " ". join (capitalized\_words) s=input("enter a string: ") print(f(s))

Output-



**Program 6 :**

**a.) Write a program that accepts a list from user. Your program should reverse the content of list and display it. Do not use reverse () method.**

**b) Find and display the largest number of a list without using built-in function max (). Your program should ask the user to input values in list from keyboard.**

**Solution :**

*Objective a) -* to reverse the content of list input by the user

l=[]

n=int(input("enter the number of elements: “)) # taking input from user for i in range(n):

l.append(int(input("enter a number: ")))

def reverse(l): # function takes list as input if(len(l)==1): # base case when list is only one item return l return reverse(l[1:])+l[0:1] # otherwise print(reverse(l)) # testing function

Output-

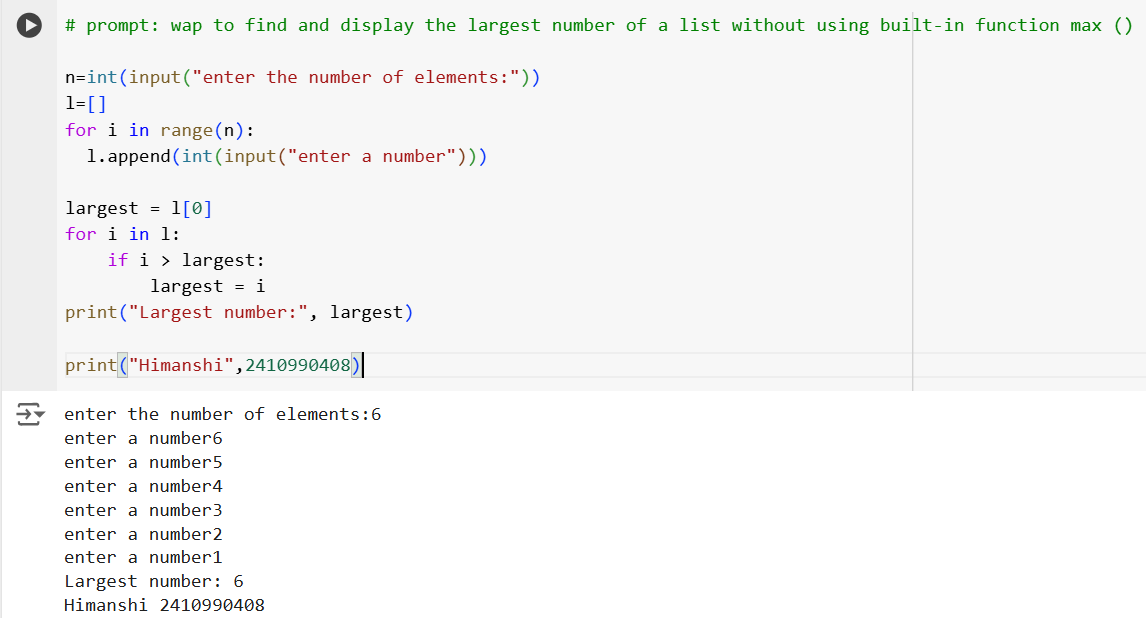


*Objective b) - t*o find and display the largest number of a list without using built-in function max () by taking input from user

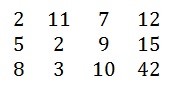
n=int(input("enter the number of elements: “)) # number of elements l=[] for i in range(n):

l.append(int(input("enter a number: “))). # input from user largest=l[0] for i in l: # checking the largest number if i>largest: largest=i print(largest)

Output-

****

**Program 7 : Find the sum of each row of matrix of size m x n. For example, for the following matrix output will be like this:**

****

**Sum of row 1 = 32**

**Sum of row 2 = 31**

**Sum of row 3 = 63**

Solution:

*Objective -* to find the sum of each row of matrix of size m x n

rows=int(input("enter number of rows: “)) #ask user for number of rows and columns

columns=int(input("enter number of columns: "))

matrix=[] # initialize an empty list to store matrix print("enter elements of matrix: “) # input values from user for matrix for i in range(rows):

row=[]

for j in range(columns):

element=int(input())

row.append(element)

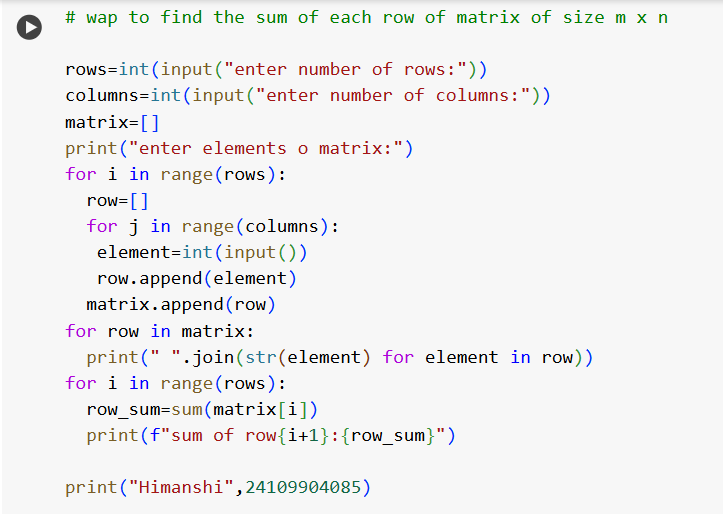
matrix.append(row)

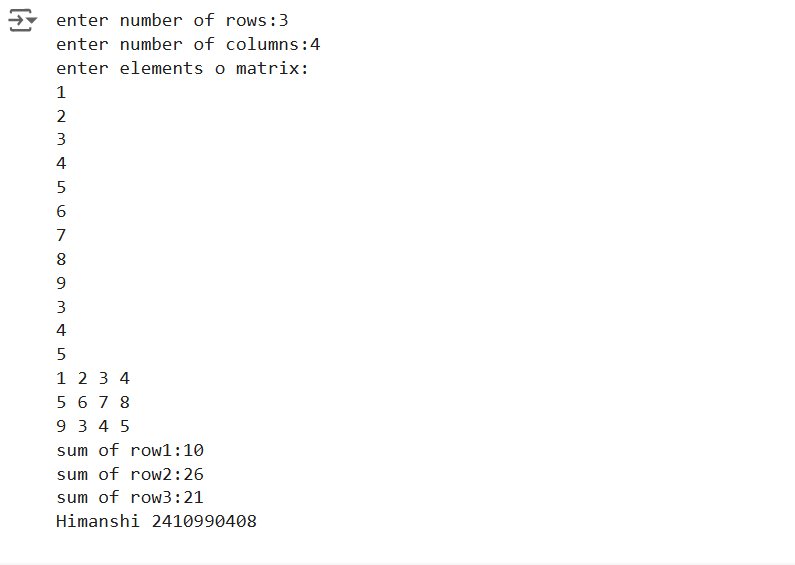
for row in matrix: # print the matrix

print(" ".join(str(element) for element in row))

for i in range(rows): # calculate and print sum of each row row\_sum=sum(matrix[i])

print(f"sum of row{i+1}:{row\_sum}”)

Output:



**Program 8 :**

**a) Write a program that reads a string from keyboard and display:**

* **The number of uppercase letters in the string.**
* **The number of lowercase letters in the string.**
* **The number of digits in the string.**
* **The number of whitespace characters in the string.**

1. [**Python Program to Find Common Characters in Two Strings.**](https://www.sanfoundry.com/python-program-check-common-letters-string/)
2. [**Python Program to Count the Number of Vowels in a String**](https://www.sanfoundry.com/python-program-count-number-vowels-string/).

Solution:

*Objective a) -* to reads a string from keyboard and display:

* The number of uppercase letters in the string.
* The number of lowercase letters in the string.
* The number of digits in the string.
* The number of whitespace characters in the string.

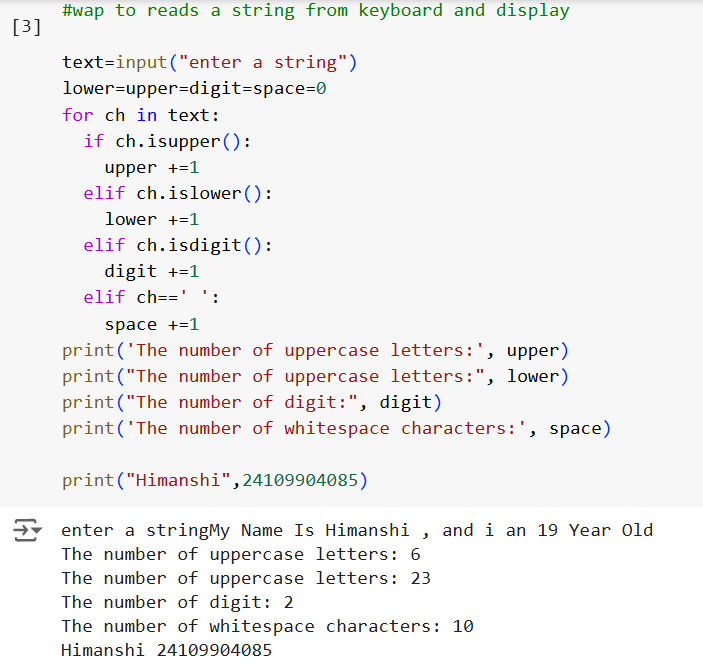
Solution:

*Objective a) -* to reads a string from keyboard and display:

* The number of uppercase letters in the string.
* The number of lowercase letters in the string.
* The number of digits in the string.
* The number of whitespace characters in the string.

text = input('Enter a string: ‘) # taking input from user lower = upper = digit = space = 0 # initialzing to zero for ch in text: # checking the constraints -if ch.isupper(): ---upper += 1 -elif ch.islower(): -------lower += 1 -elif ch.isdigit(): ---digit += 1 -elif ch == ' ': ---space += 1 print('The number of uppercase letters:', upper) print('The number of lowercase letters:', lower) print('The number of digits:', digit) print('The number of whitespace characters:', space)

Output-

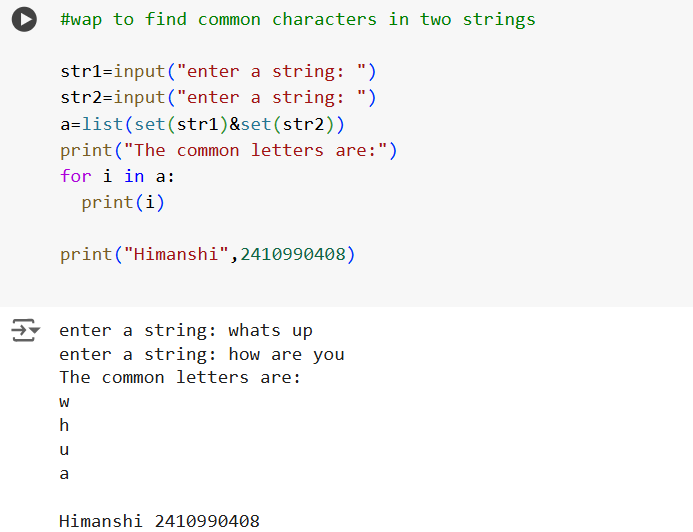


*Objective b) -* to find common characters in two strings input from user

str1=input("enter a string: “) # taking input from user str2=input("enter a string: ")

a=list(set(str1)&set(str2)) # converting list into sets and finding common characters print("The common letters are:") for for i in a: ---------print(i)

Output-



*Objective c) -* to count number of vowels in a string input by user

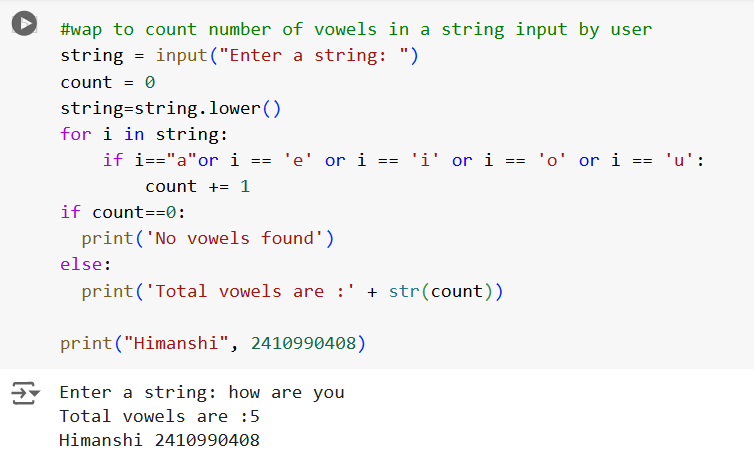
String = input('Enter the string :’) # taking input from user count = 0 # initialising to zero

String = String.lower() # keeping string in lower case for i in String: --if i == 'a' or i == 'e' or i == 'i' or i == 'o' or i == 'u':

count+=1

if count == 0: # checking if vowel found --print('No vowels found') else: ------- print('Total vowels are :' + str(count))

Output-



**Program 9 :**

1. **Write a Python program to check if a specified element presents in a tuple of tuples. Original list:**

**((‘Red’ ,’White’ , ‘Blue’),(‘Green’, ’Pink’ , ‘Purple’), (‘Orange’, ‘Yellow’, ‘Lime’)) Check if White present in said tuple of tuples!**

**True**

**Check if Olive present in said tuple of tuples!**

**False**

1. **Write a Python program to remove an empty tuple(s) from a list of tuples.**

**Sample data: [(), (), ('',), ('a', 'b'), ('a', 'b', 'c'), ('d')] Expected output: [('',), ('a', 'b'), ('a', 'b', 'c'), ‘d']**

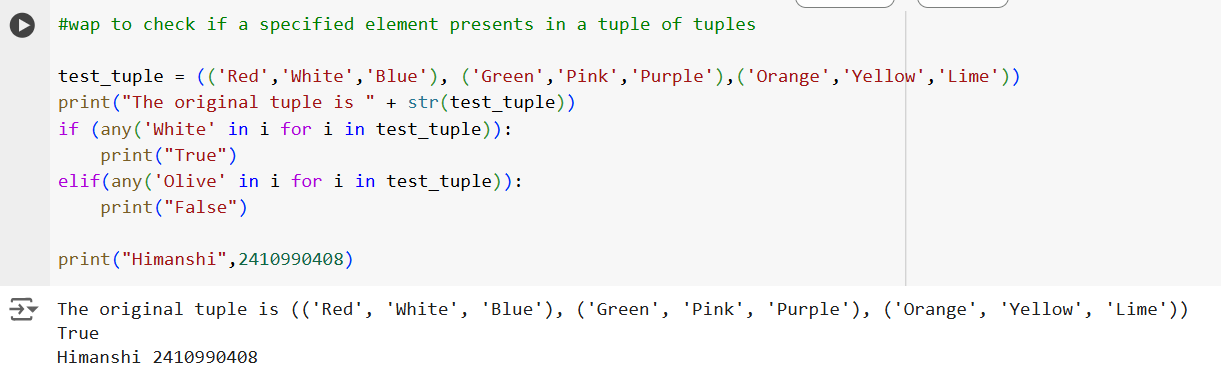
Solution- *Objective a) -* to check if a specified element presents in a tuple of tuples

test\_tuple = (('Red','White','Blue'), ('Green','Pink','Purple'),

('Orange','Yellow','Lime')) print("The original tuple is " + str(test\_tuple)) if (any('White' in i for i in test\_tuple)):

print("True") elif(any('Olive' in i for i in test\_tuple)): ----------------- print(“False")

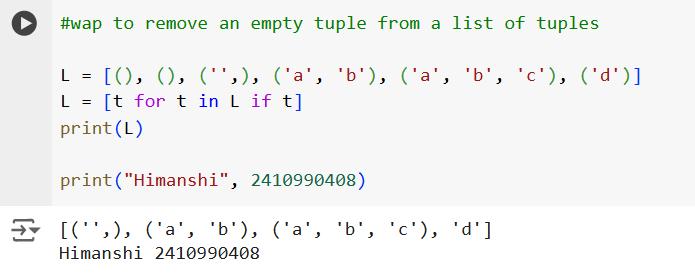
Output-



*Objective b) -* to remove an empty tuple from a list of tuples

L = [(), (), ('',), ('a', 'b'), ('a', 'b', 'c'), (‘d')] # create a list with tuples L = [t for t in L if t] print(L) # Use a list comprehension to filter out the empty tuples by checking if each tuple 't' in 'L' is not empty

Output-



**Program 10 : Write a Program in Python to Find the Differences Between Two Lists Using Sets**.

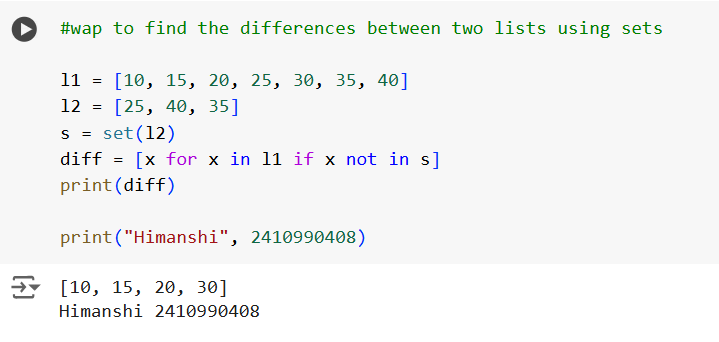
**Solution :**

*Objective -* to find the differences between two lists using sets

l1 = [10, 15, 20, 25, 30, 35, 40] l2 = [25, 40, 35] s = set(l2)

diff = [x for x in l1 if x not in s] print(diff)

Output-



**Program 11 :**

1. **Write a Python program Remove duplicate values** **across Dictionary Values.**

**Input : test\_dict = {‘Manjeet’: [1], ‘Akash’: [1, 8, 9]}**

**Output : {‘Manjeet’: [], ‘Akash’: [8, 9]}**

**Input : test\_dict = {‘Manjeet’: [1, 1, 1], ‘Akash’: [1, 1, 1]}**

**Output : {‘Manjeet’: [], ‘Akash’: []}**

1. **Write a Python program to Count the frequencies in a list using dictionary in Python. Input : [1, 1, 1, 5, 5, 3, 1, 3, 3, 1,4, 4, 4, 2, 2, 2, 2] Output :** 
   1. **: 5**
   2. **: 4**
   3. **: 3**
   4. **: 3**
   5. : **2**

**Explanation : Here 1 occurs 5 times, 2 occurs 4 times and so on**

**Solution :**

*Objective a) -* to remove duplicate values across dictionary values

test\_dict = {'hi': 10, 'i': 15, 'am': 20, 'famous': 10, 'jk': 20}

print("The original dictionary is : " + str(test\_dict)) temp = []

res = dict()

for key, val in test\_dict.items():

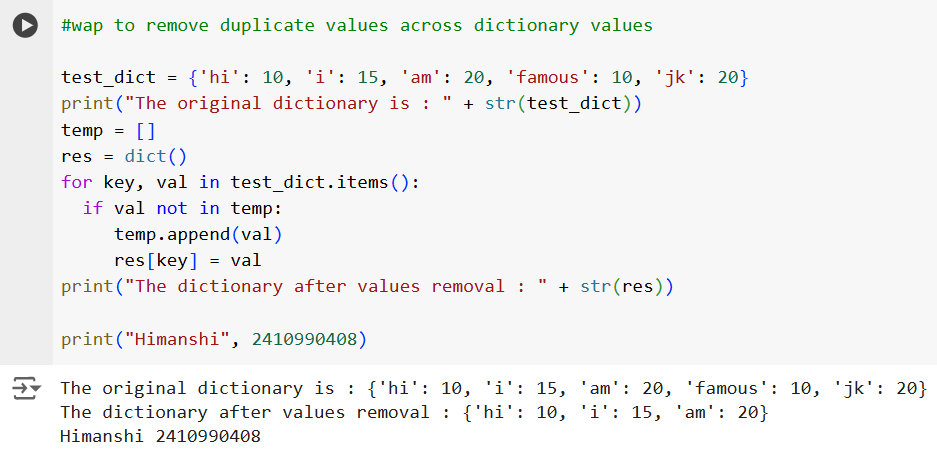
if val not in temp:

temp.append(val)

res[key] = val

print("The dictionary after values removal : " + str(res))

Output-



*Objective b) -* to count the frequencies in a list using dictionary in Python

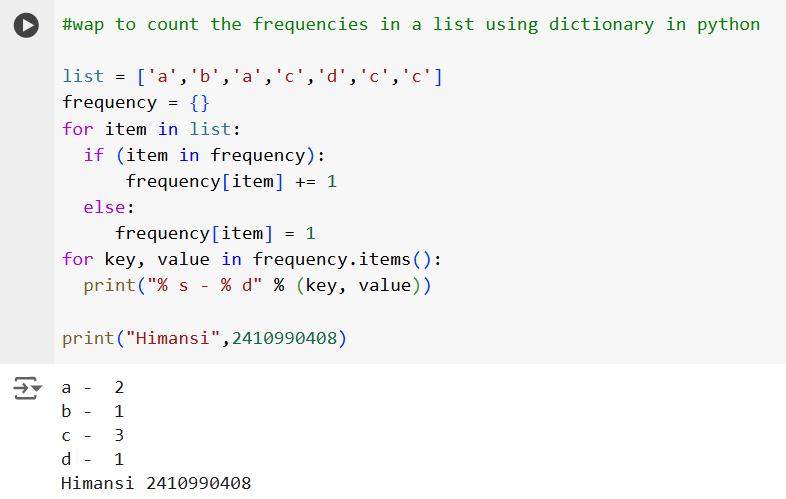
list = ['a','b','a','c','d','c','c']

frequency = {}

for item in list: # checking the count of frequencies ---if (item in frequency): ------------------------frequency[item] += 1 ----else:

frequency[item] = 1 for key, value in frequency.items(): # using key value dictionary print("% s - % d" % (key, value))

Output-



**Program 12 :**

**a) Write a Python Program to Capitalize First Letter of Each Word in a File.**

**b.) Write a Python Program to Print the Contents of File in Reverse Order**.

**Solution:**

*Objective a) -**to Capitalize First Letter of Each Word in a File.*

**Program 13 :**

**WAP to catch an exception and handle it using try and except code blocks**.

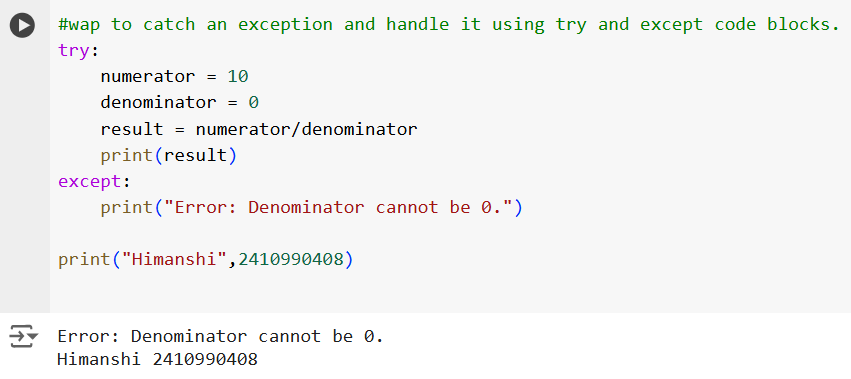
**Solution :**

*Objective -* to catch an exception and handle it using try and except code blocks.

try:

numerator = 10 ---denominator = 0 -------result = numerator/denominator ---print(result) except: -----------print("Error: Denominator cannot be 0.”)

Output-



**Program 14 :**

**Write a Python Program to Append, Delete and Display Elements of a List using Classes**.

**Solution :**

*Objective -* to append , delete and display elements of a list using classes

class check(): # class called check created -----def \_\_init\_\_(self): # \_int\_() method used to initialize values of class ------self.n=[] def add(self,a): # methods for add, remove and display elements defined ---return self.n.append(a) def remove(self,b): --------------self.n.remove(b) def dis(self):

return (self.n) obj=check()

choice=1 while choice!=0:

print("0. Exit")

print("1. Add")

print("2. Delete")

print("3. Display”) # menu is printed and choice is taken from user =choice=int(input("Enter choice: “))

=if choice==1:

n=int(input("Enter number to append: “)) # object for class is created -- obj.add(n)

print("List:",obj.dis())

elif choice==2:

n=int(input("Enter number to remove: "))

obj.remove(n) # using object respective method is called ------print("List: “,obj.dis()) # according to choice taken from user elif choice==3:

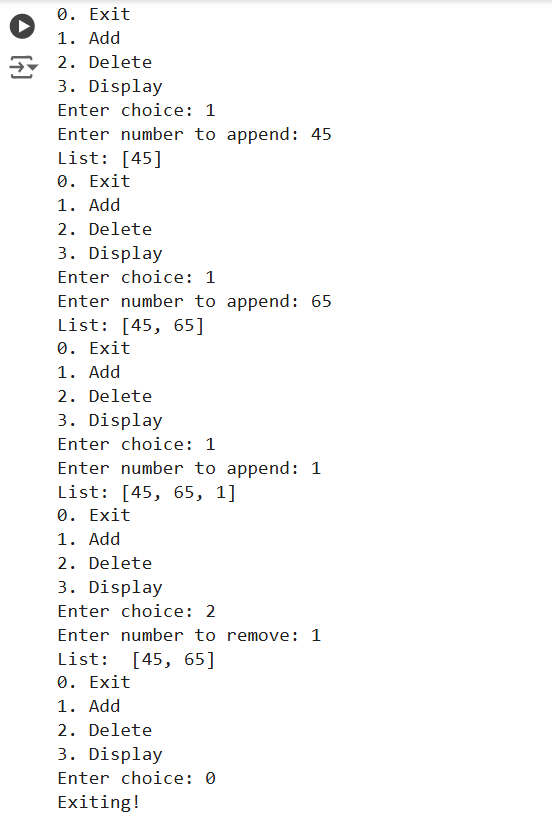
print("List: ",obj.dis()) elif choice==0:

print("Exiting!") else:

print("Invalid choice!!")

print() # final list printed

Output-



**Program 15 :**

**Write a** [**Python Program to Find the Area and Perimeter of the Circle using Class.**](https://www.sanfoundry.com/python-program-class-compute-area-perimeter-circle/)

**Solution :**

*Objective -* to find area and perimeter of circle using class by input from user

import math class circle():

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

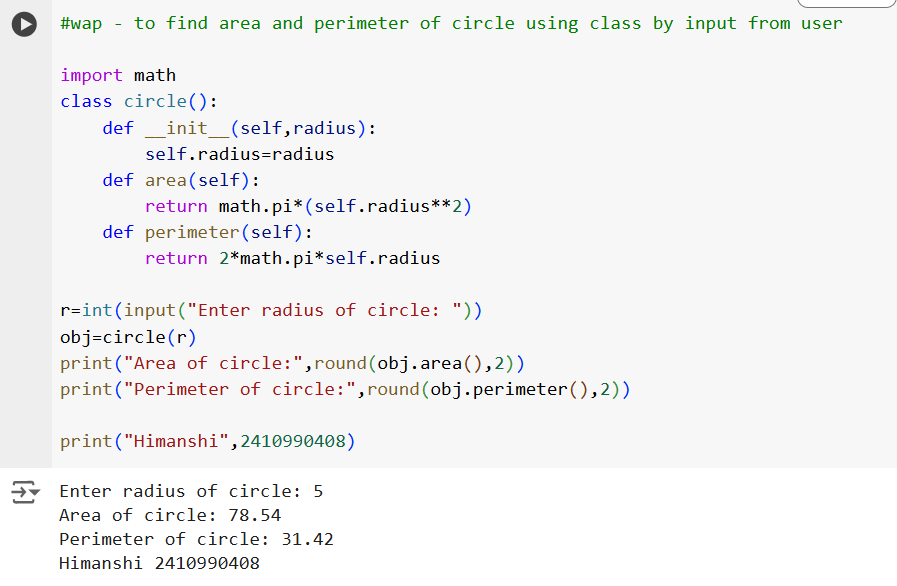
self.radius=radius def area(self):

return math.pi\*(self.radius\*\*2) def perimeter(self):

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

r=int(input("Enter radius of circle: ")) obj=circle(r) print("Area of circle:",round(obj.area(),2)) print("Perimeter of circle:”,round(obj.perimeter(),2))

Output-



**Program 16 :**

**Create an interactive application using Python's Tkinter library for graphics programming**.

**Solution :**

*Objective -* to create an interactive application using Python's Tkinter library for graphics programming