**PYTHON PROGRAM 1:**

**Python program to Count the Number of Vowels Present in a String using Sets Step1: Get a string Step2: find the no of occurrence of all vowels in the string using data structure set.**

s = input("Enter The String: ")

v = set("aeiouAEIOU")

c = 0

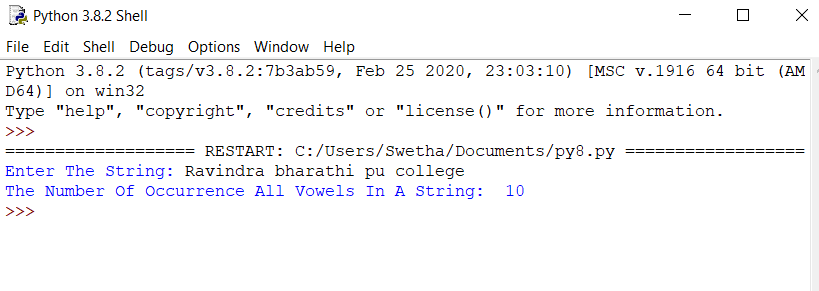
for i in s:

if i in v:

c += 1

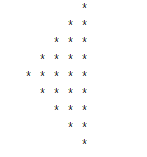
print("The Number Of Occurrence All Vowels In A String: ", c)

**OUTPUT:**



**PYTHON PROGRAM 2:**

**Write a python program to print the pattern for n value.**

  
rows = 5

k = 2 \* rows - 2

for i in range(rows, -1, -1):

for j in range(k, 0, -1):

print(end=" ")

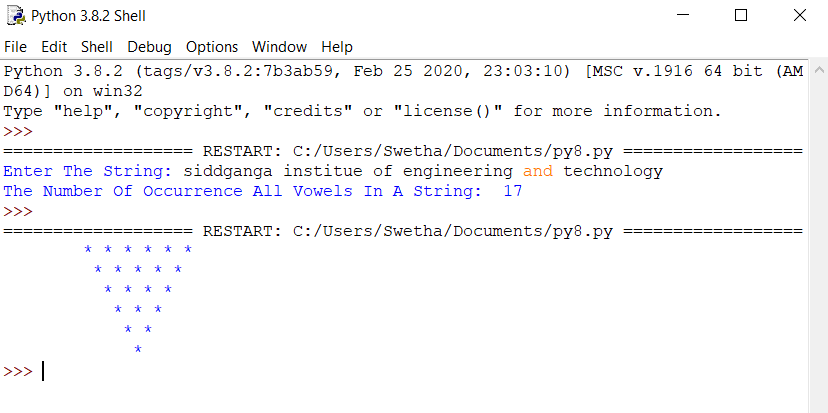
k = k + 1

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

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

print("")

**OUTPUT:**



**PYTHON PROGRAM 3:**

**A magic square of order n is an arrangement of n^2 numbers, usually distinct integers, in a square, such that the n numbers in all rows, all columns, and both diagonals sum to the same constant. A magic square contains the integers from 1 to n^2.**

def gSqua(n):

magic\_square = [[0 for x in range(n)] for y in range(n)]

i = n / 2

j = n - 1

num = 1

while num <= (n \* n):

if i == -1 and j == n:

j = n - 2

i = 0

else:

if j == n:

j = 0

if i < 0:

i = n - 1

if magic\_square[int(i)][int(j)]:

j = j - 2

i = i + 1

continue

else:

magic\_square[int(i)][int(j)] = num

num = num + 1

j = j + 1

i = i - 1

print("Magic Squre For n =", n)

print('Sum Of Each Row Or Column Or Diagonal: ', n \* (n \* n + 1) / 2, "\n")

for i in range(0, n):

for j in range(0, n):

print('%2d ' % (magic\_square[i][j]), end='')

if j == n - 1:

print()

n = int(input("Enter The Number For Magic Square: "))

gSqua(n)

**OUTPUT:**

