Practical No :- 01

Aim :- Write a python program for Sum of squares of first n natural numbers.

Program :-

# Return the sum of

# square of first n

# natural numbers

def squaresum(n):

# Iterate i from 1

# and n finding

# square of i and

# add to sum.

sm = 0

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

sm = sm + (i \* i)

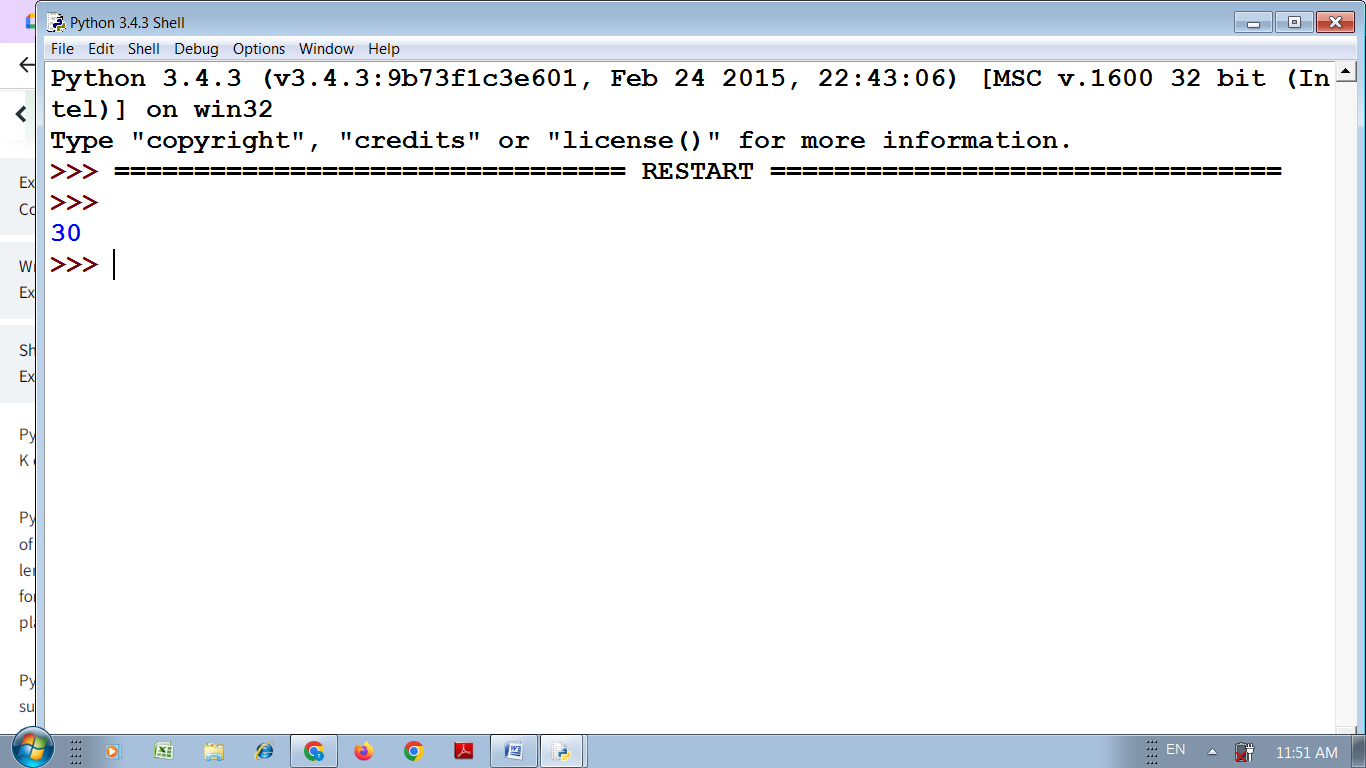
return sm

# Driven Program

n = 4

print(squaresum(n))

Output :-



Practical No :- 02

Aim :- Python Program to merge two files into a third file.

Program :-

data = data2 = "";

# Reading data from file1

with open('file1.txt') as fp:

    data = fp.read()

# Reading data from file2

with open('file2.txt') as fp:

    data2 = fp.read()

# Merging 2 files

# To add the data of file2

# from next line

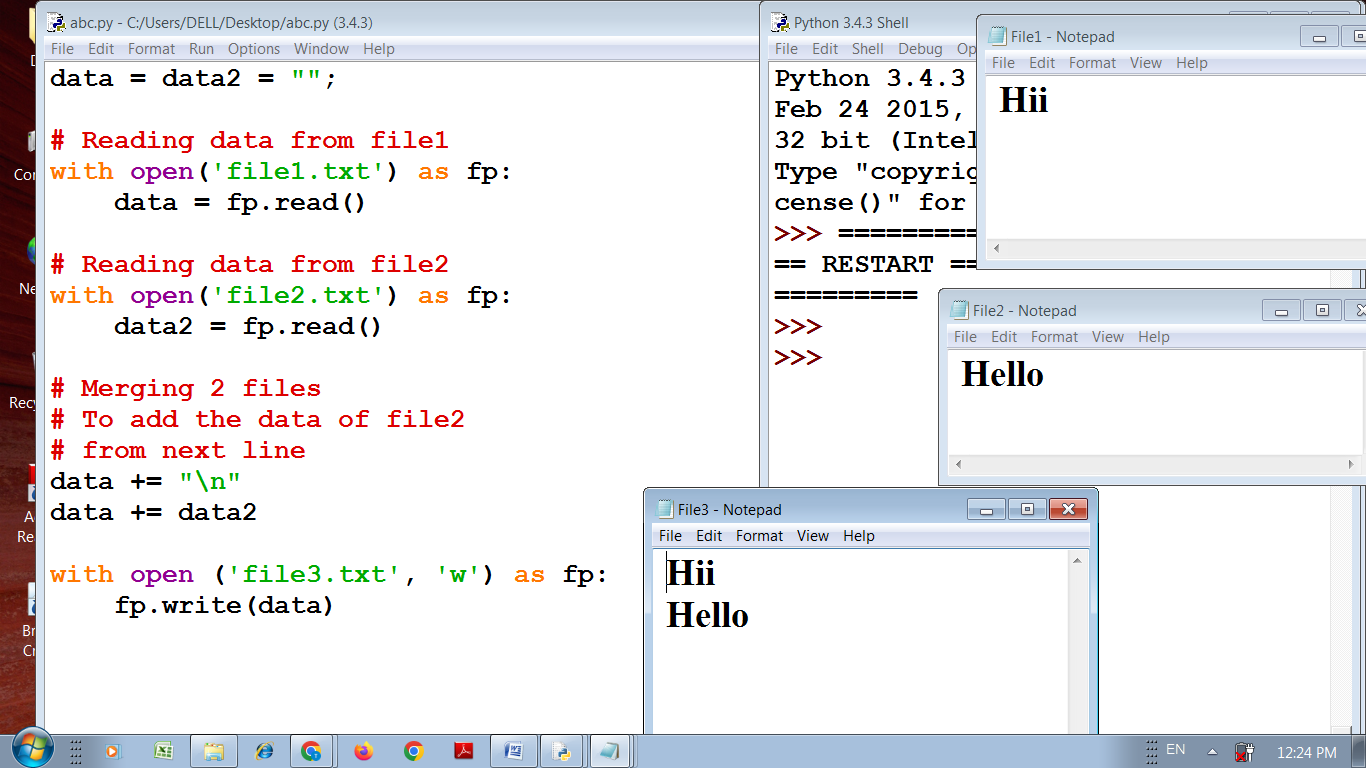
data += "\n"

data += data2

with open ('file3.txt', 'w') as fp:

    fp.write(data)

Output :-



Practical No :- 03

Aim :- Write a python program to calculate simple Interest and compound Interest.

Program :-

# Simple and Compound Interest

# Reading principal amount, rate and time

principal = float(input('Enter amount: '))

time = float(input('Enter time: '))

rate = float(input('Enter rate: '))

# Calcualtion

simple\_interest = (principal\*time\*rate)/100

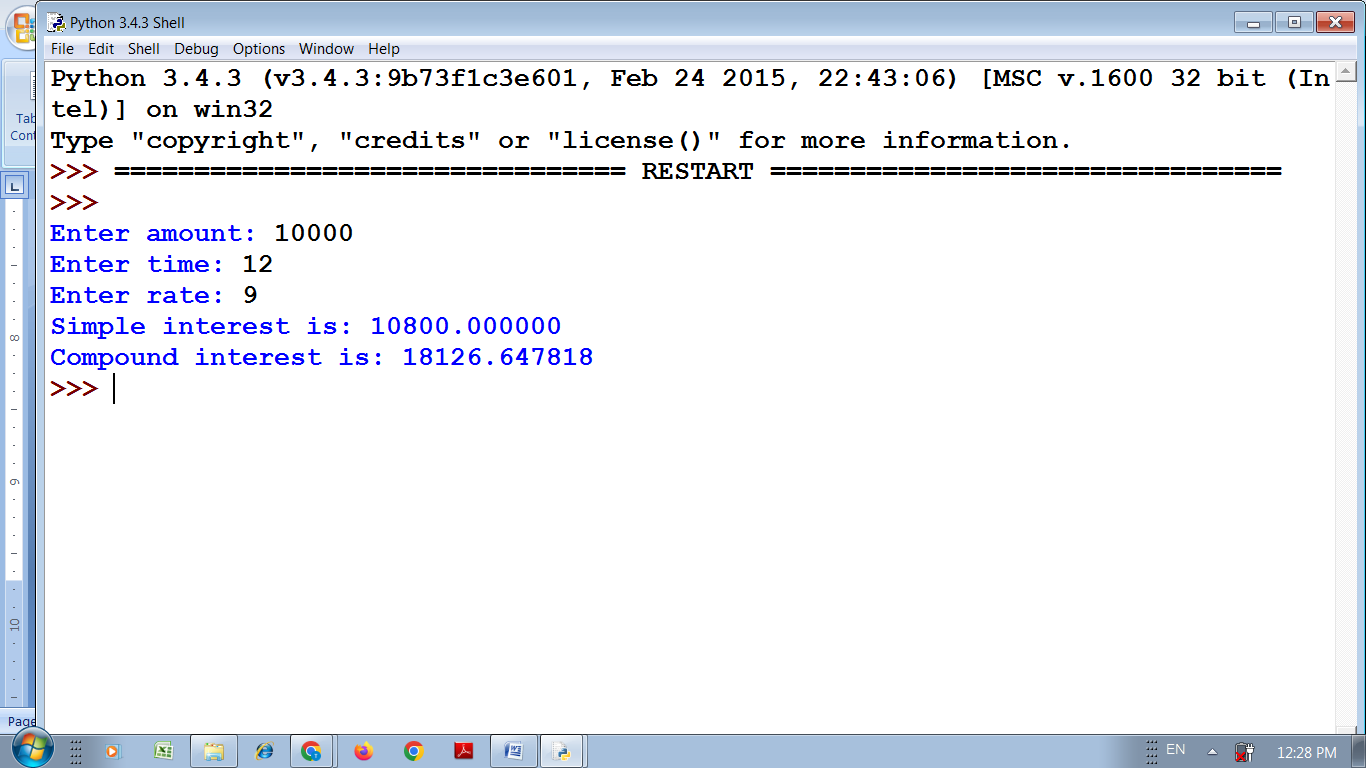
compound\_interest = principal \* ( (1+rate/100)\*\*time - 1)

# Displaying result

print('Simple interest is: %f' % (simple\_interest))

print('Compound interest is: %f' %(compound\_interest))

Output :-



Practical No :- 04

Aim :- Python Program to demonstrate operations on List.

Program :-

# Python code for various list operation

# declaring a list of integers

iList = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

# List slicing operations

# printing the complete list

print('iList: ',iList)

# printing first element

print('first element: ',iList[0])

# printing fourth element

print('fourth element: ',iList[3])

# printing list elements from 0th index to 4th index

print('iList elements from 0 to 4 index:',iList[0: 5])

# printing list -7th or 3rd element from the list

print('3rd or -7th element:',iList[-7])

# appending an element to the list

iList.append(111)

print('iList after append():',iList)

# finding index of a specified element

print('index of \'80\': ',iList.index(80))

# sorting the elements of iLIst

iList.sort()

print('after sorting: ', iList);

# popping an element

print('Popped elements is: ',iList.pop())

print('after pop(): ', iList);

# removing specified element

iList.remove(80)

print('after removing \'80\': ',iList)

# inserting an element at specified index

# inserting 100 at 2nd index

iList.insert(2, 100)

print('after insert: ', iList)

# counting occurances of a specified element

print('number of occurences of \'100\': ', iList.count(100))

# extending elements i.e. inserting a list to the list

iList.extend([11, 22, 33])

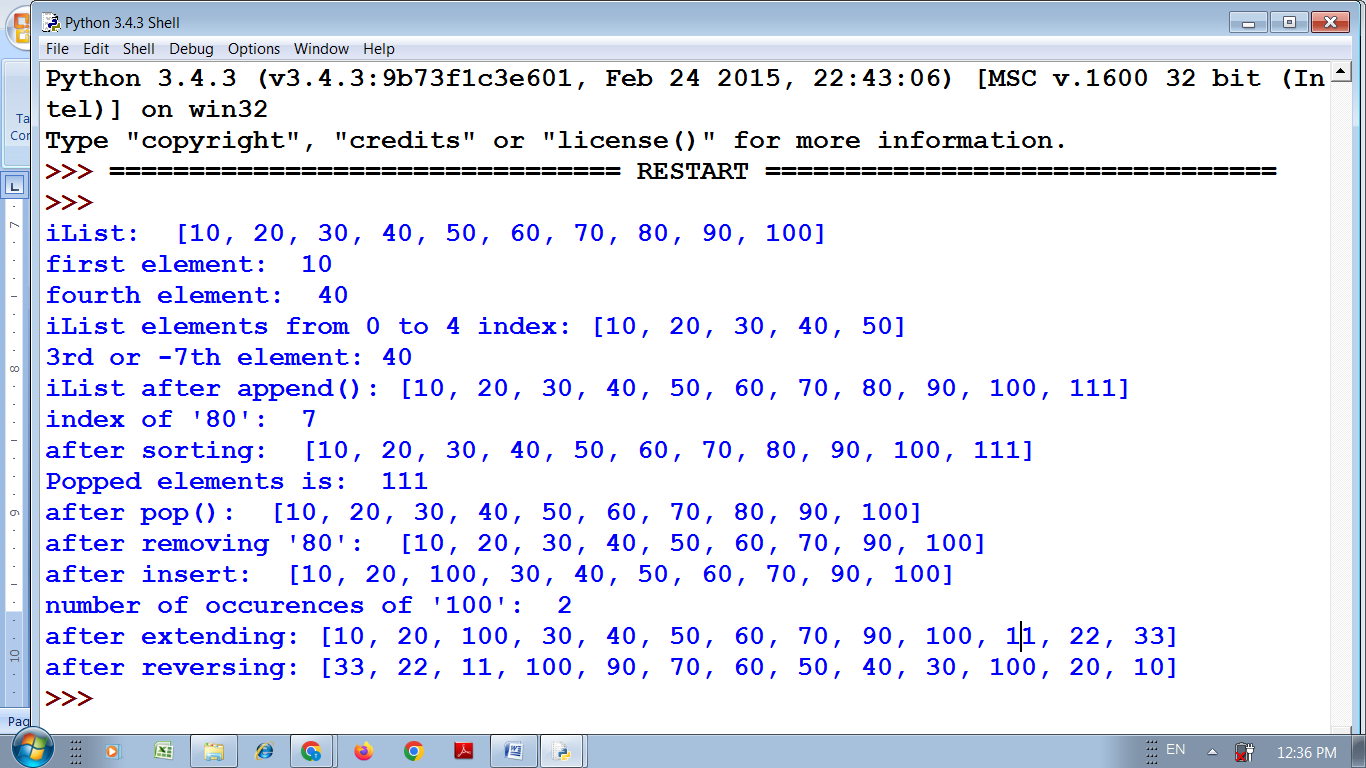
print('after extending:', iList)

#reversing the list

iList.reverse()

print('after reversing:', iList)

Output :-



Practical No :- 05

Aim :- Write a python program to make use of following functions of math module: exp(),floor(),factorial(), sqrt(), pow(), cos().

Program :-

# Python3 code to demonstrate

# the working of exp()

import math

# initializing the value

test\_int = 4

test\_neg\_int = -3

test\_float = 0.00

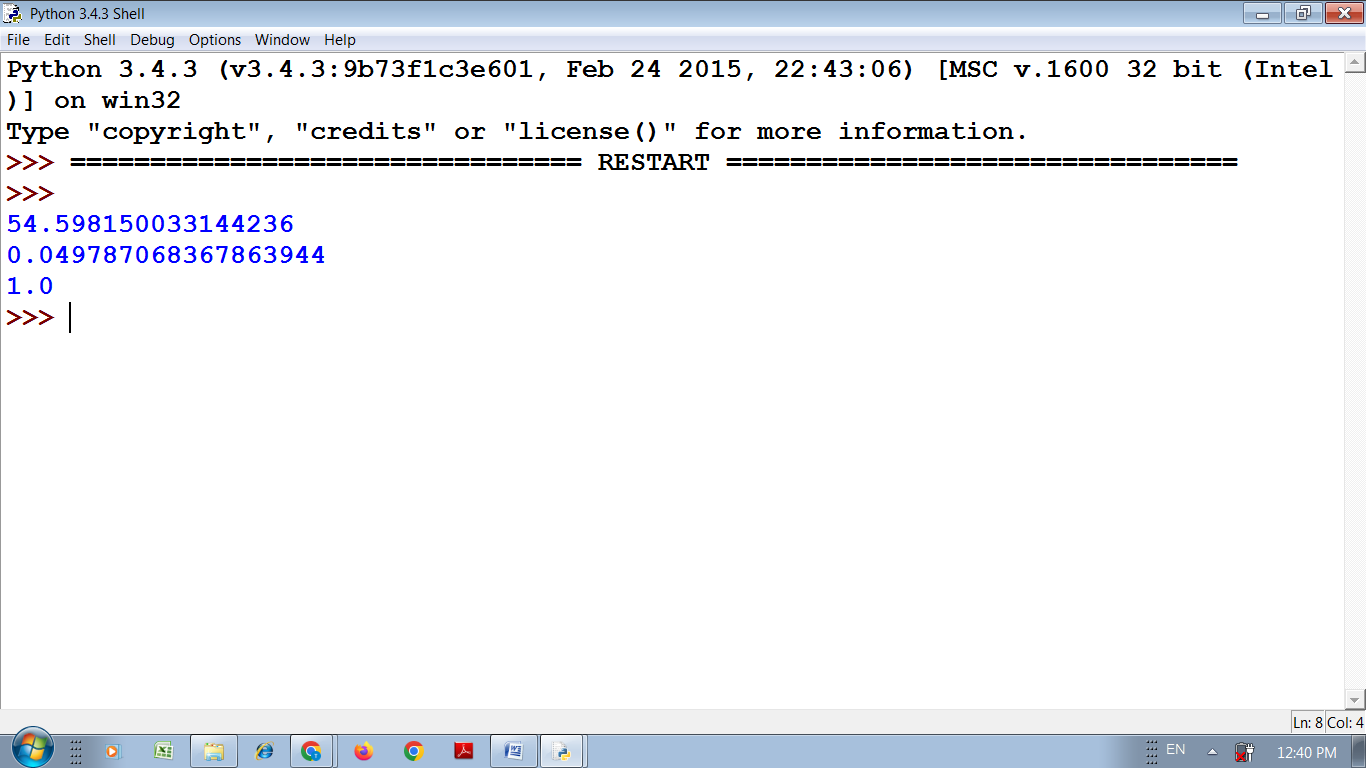
# checking exp() values

# with different numbers

print (math.exp(test\_int))

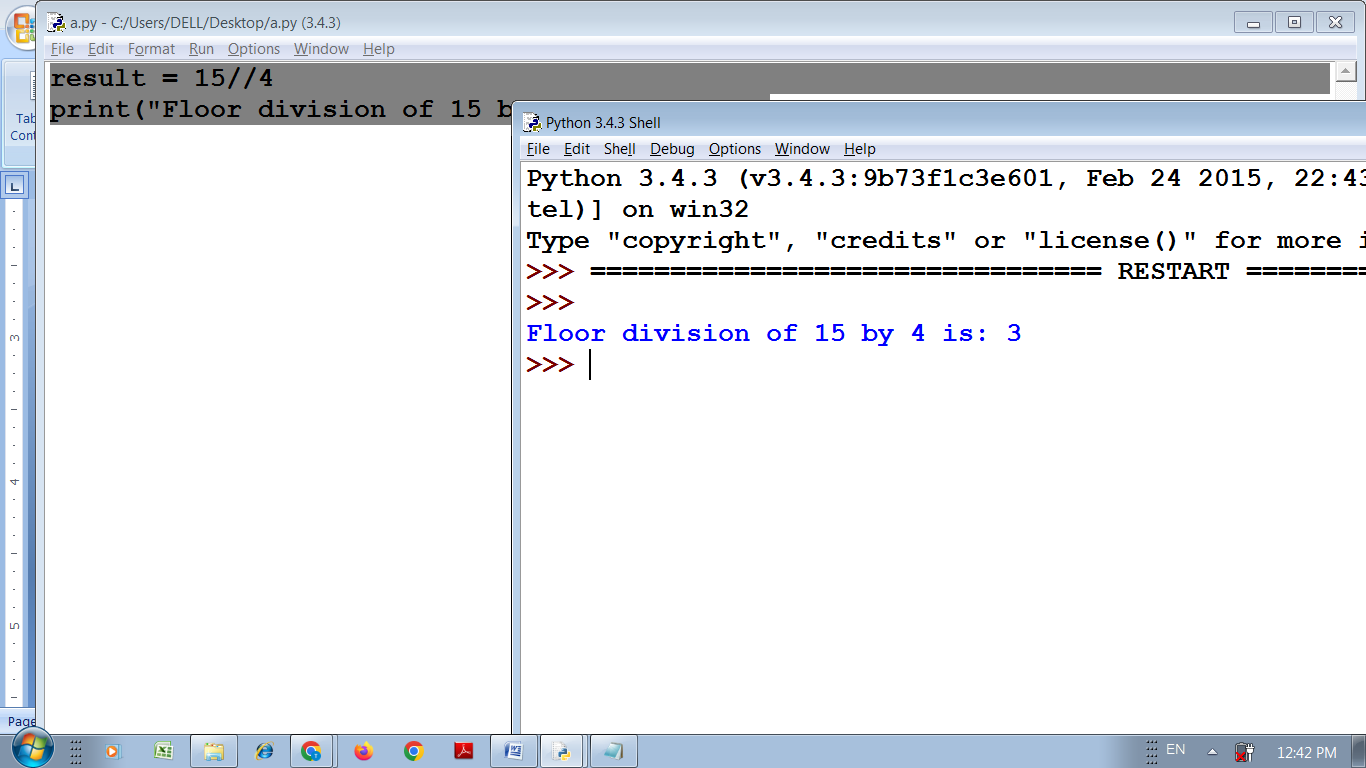
print (math.exp(test\_neg\_int))

print (math.exp(test\_float))



result = 15//4

print("Floor division of 15 by 4 is:",result)



# Python program to find

# factorial of given number

import math

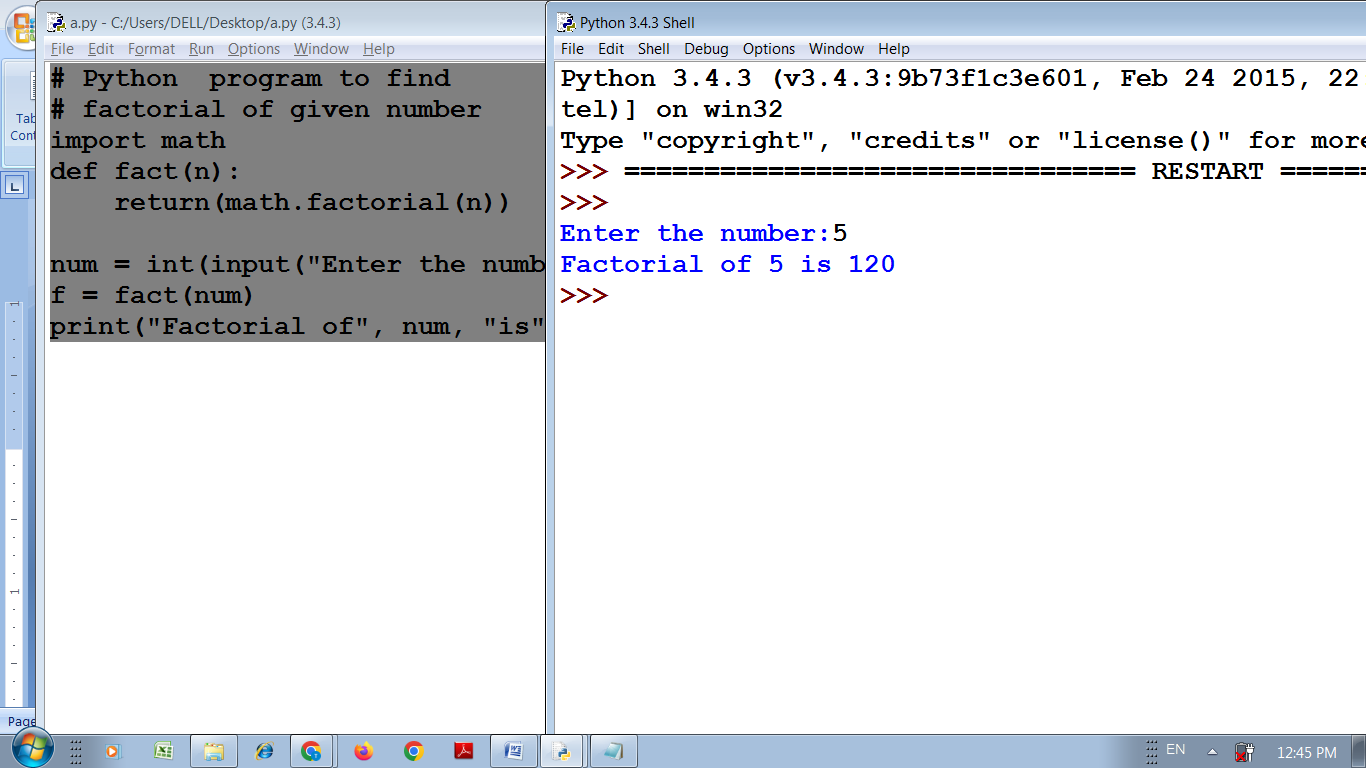
def fact(n):

return(math.factorial(n))

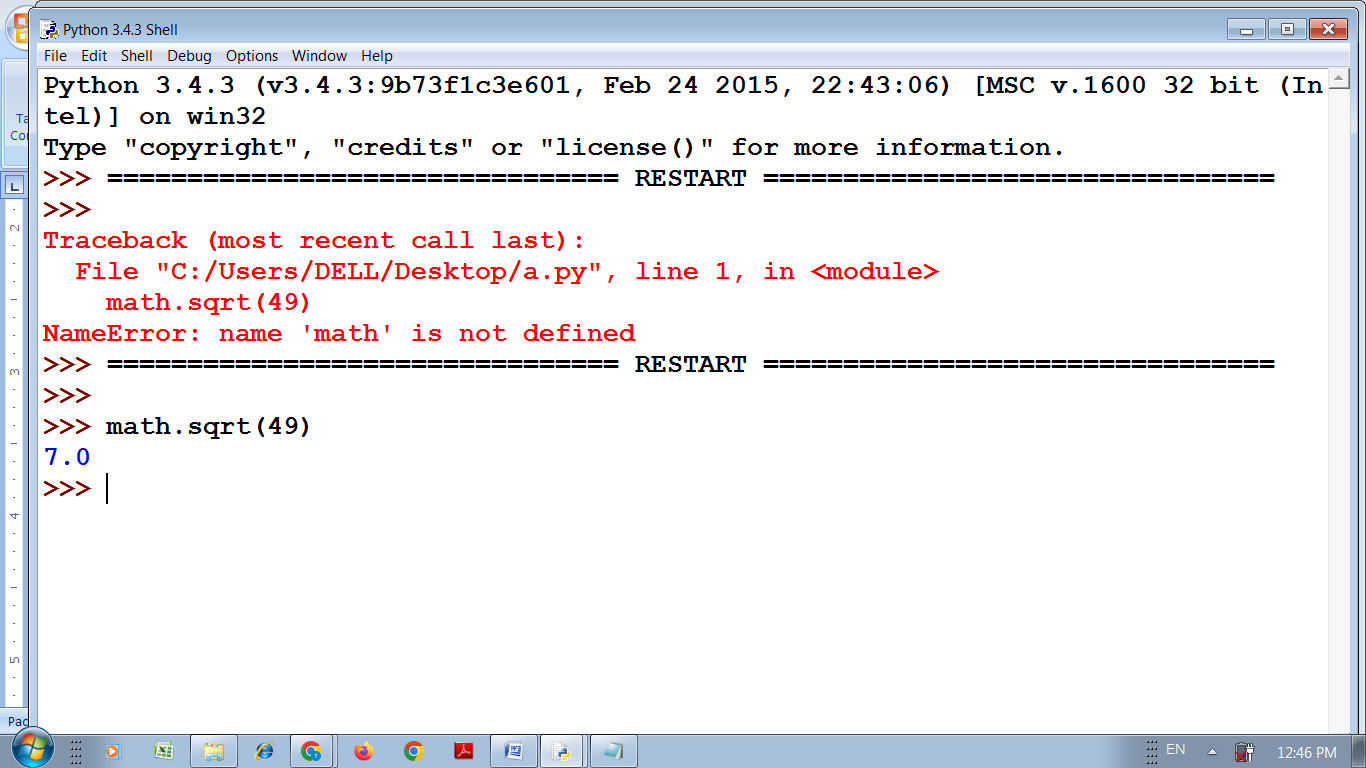
num = int(input("Enter the number:"))

f = fact(num)

print("Factorial of", num, "is", f)



math.sqrt(49)



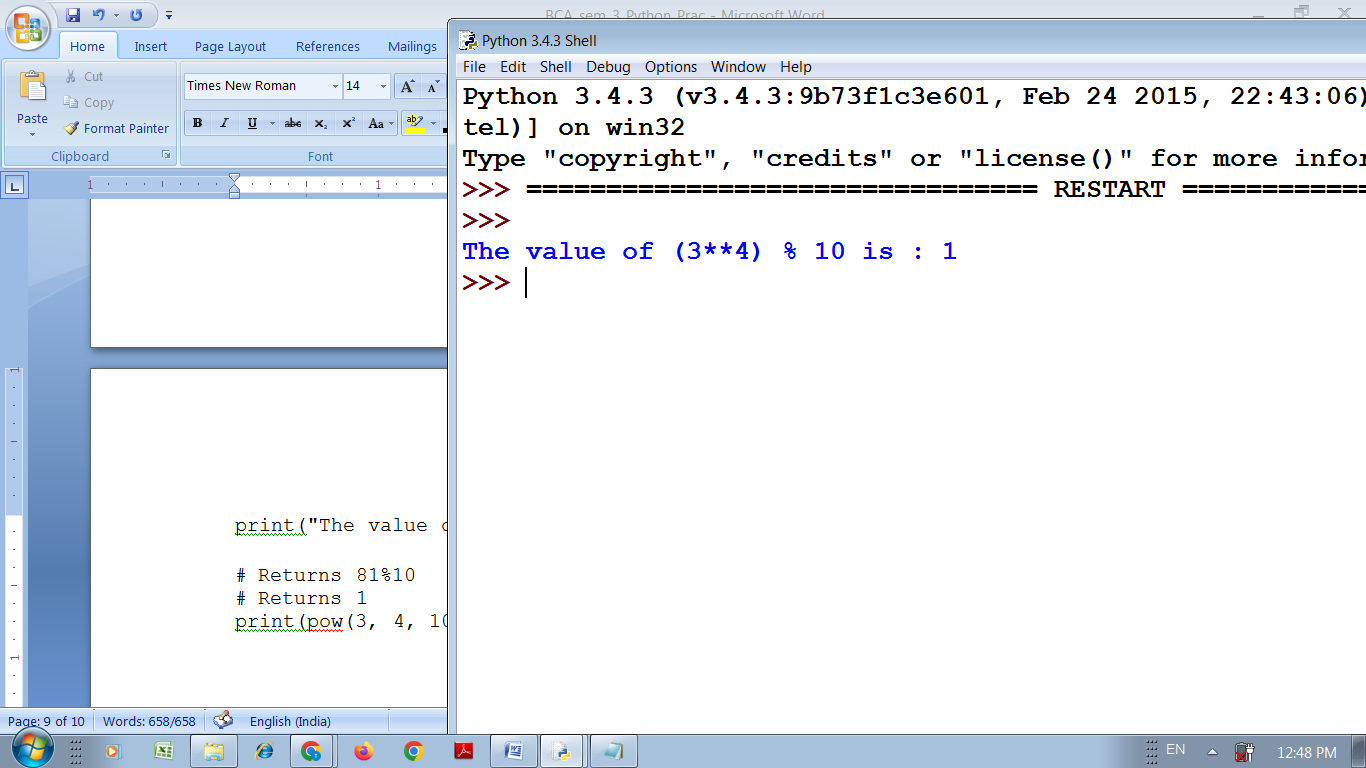
# Python code to demonstrate pow()

print("The value of (3\*\*4) % 10 is : ", end="")

# Returns 81%10

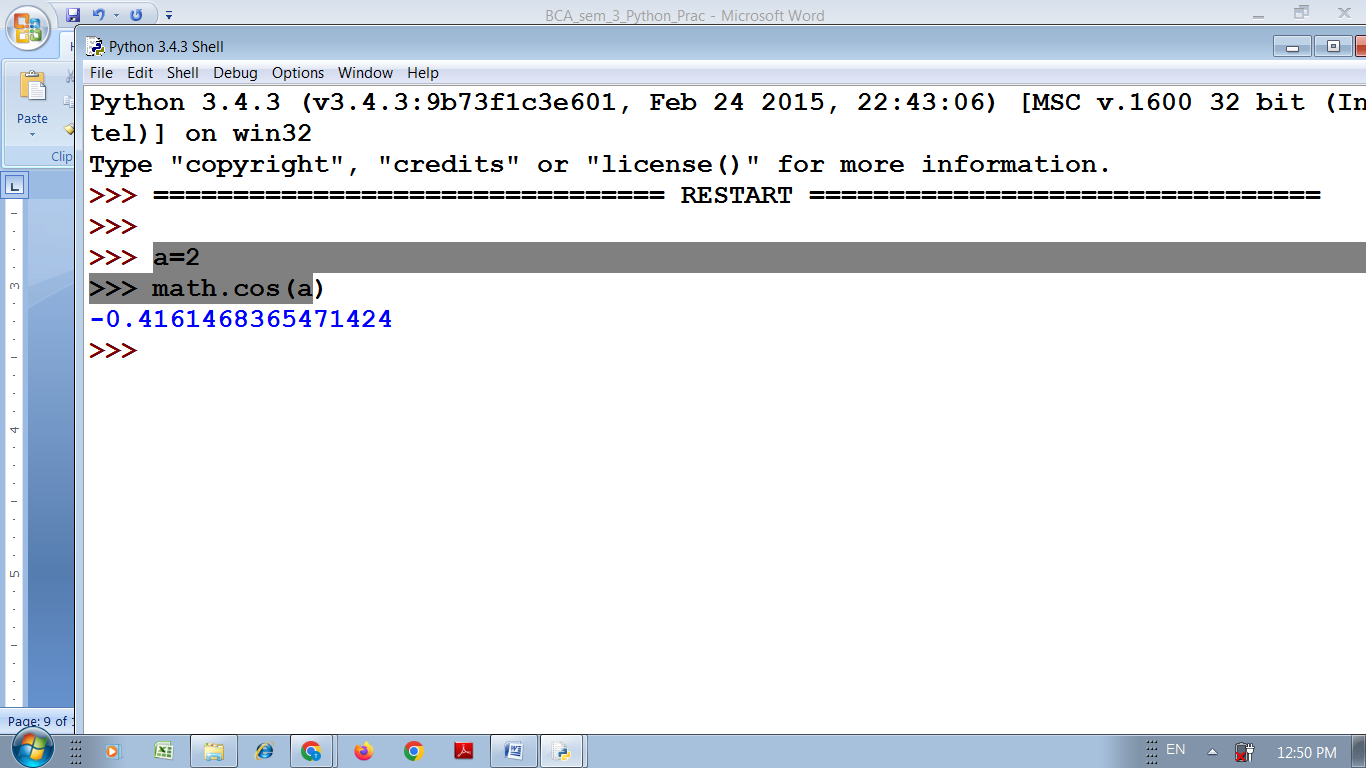
# Returns 1

print(pow(3, 4, 10))



a=2

math.cos(a)



Practical No :- 06

Aim :- Write a python program to demonstrate use of recursive function.

Program :-

def factorial(x):

"""This is a recursive function

to find the factorial of an integer"""

if x == 1:

return 1

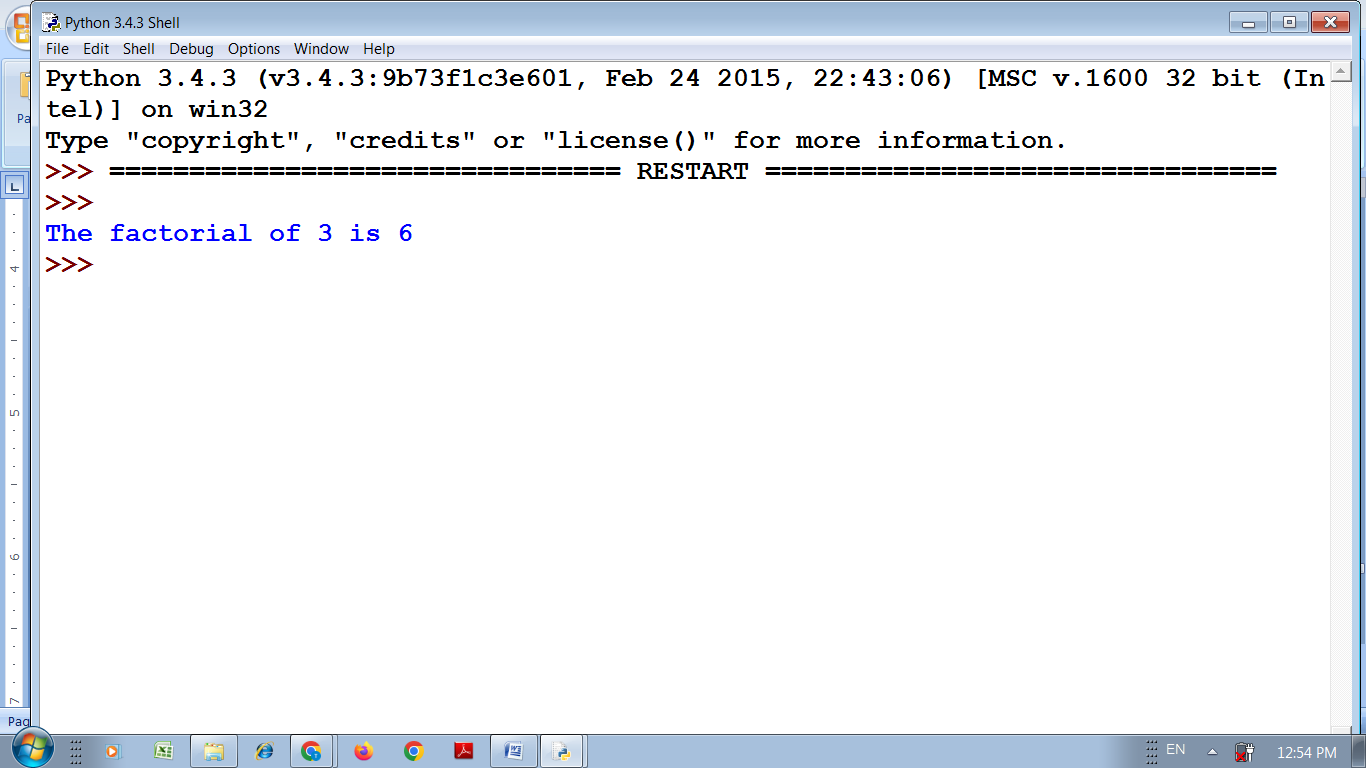
else:

return (x \* factorial(x-1))

num = 3

print("The factorial of", num, "is", factorial(num))

Output :-



Practical No :- 07

Aim :- Python Program to demonstrate operations on Tuples.

Program :-

# Creating an empty Tuple

Tuple1 = ()

print("Initial empty Tuple: ")

print(Tuple1)

# Creating a Tuple

# with the use of string

Tuple1 = ('Geeks', 'For')

print("\nTuple with the use of String: ")

print(Tuple1)

# Creating a Tuple with

# the use of list

list1 = [1, 2, 4, 5, 6]

print("\nTuple using List: ")

print(tuple(list1))

# Creating a Tuple

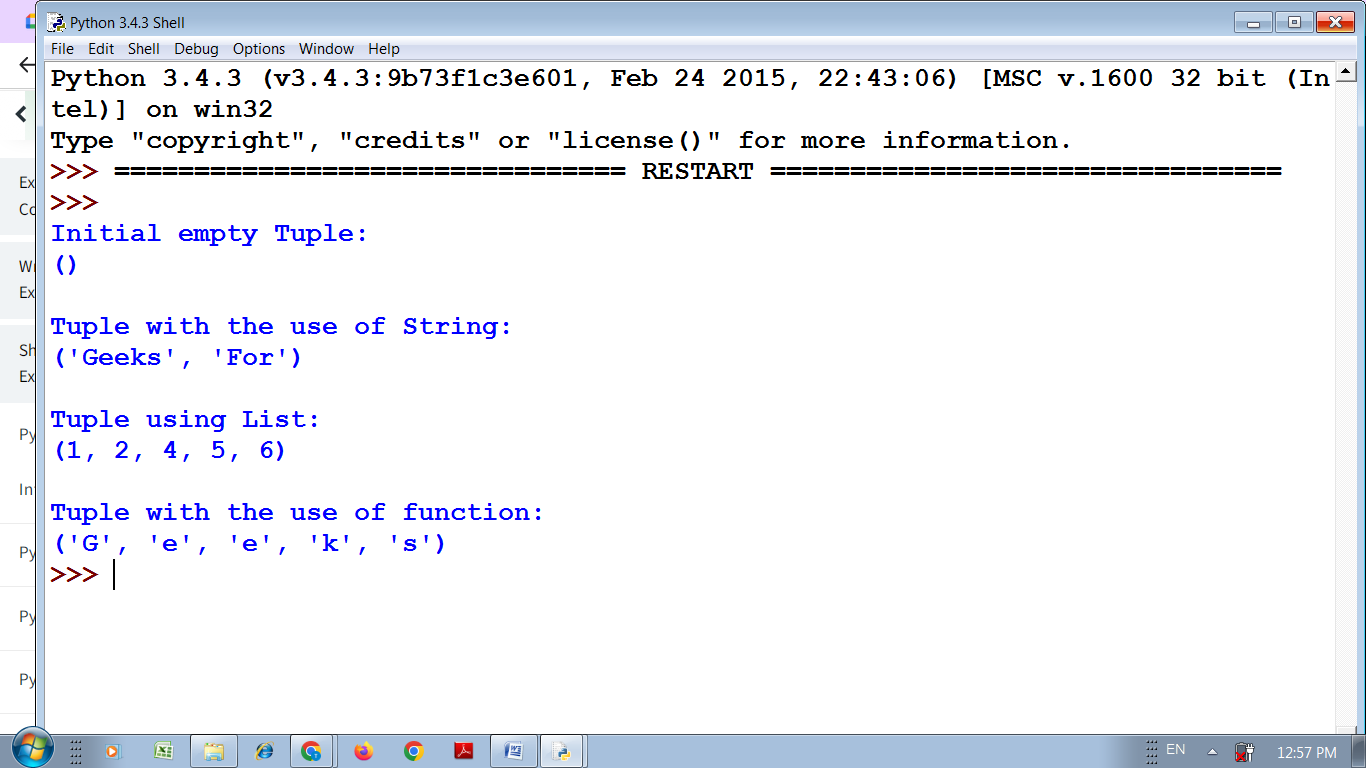
# with the use of built-in function

Tuple1 = tuple('Geeks')

print("\nTuple with the use of function: ")

print(Tuple1)

Output :-



Practical No :- 08

Aim :- Write a python program using matplotlib to create Histogram.

Program :-

from matplotlib import pyplot as plt

import numpy as np

# Creating dataset

a = np.array([22, 87, 5, 43, 56,

              73, 55, 54, 11,

              20, 51, 5, 79, 31,

              27])

# Creating histogram

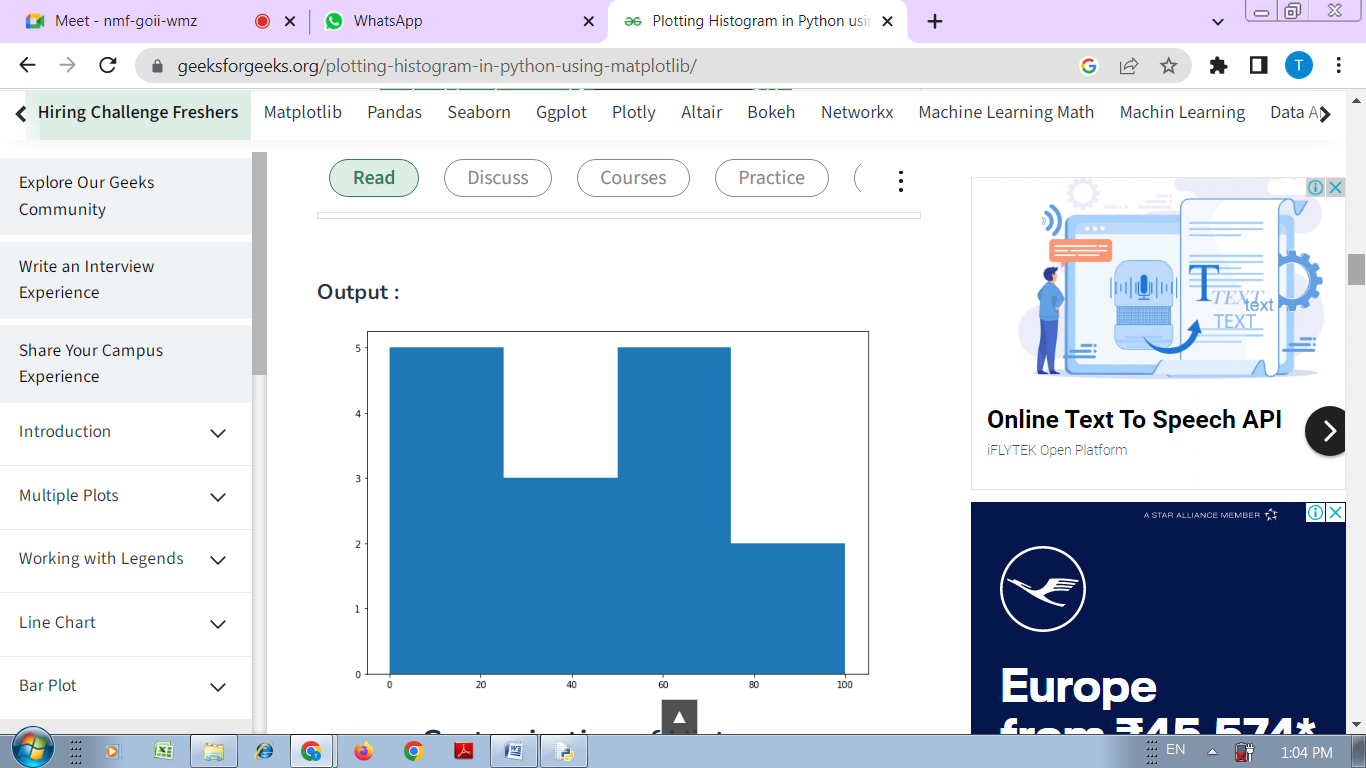
fig, ax = plt.subplots(figsize =(10, 7))

ax.hist(a, bins = [0, 25, 50, 75, 100])

# Show plot

plt.show()

Output :-



Practical No :- 09

Aim :- Python program to remove punctuation from string.

Program :-

# define punctuation

punctuations = '''!()-[]{};:'"\,<>./?@#$%^&\*\_~'''

my\_str = "Hello!!!, he said ---and went."

# To take input from the user

# my\_str = input("Enter a string: ")

# remove punctuation from the string

no\_punct = ""

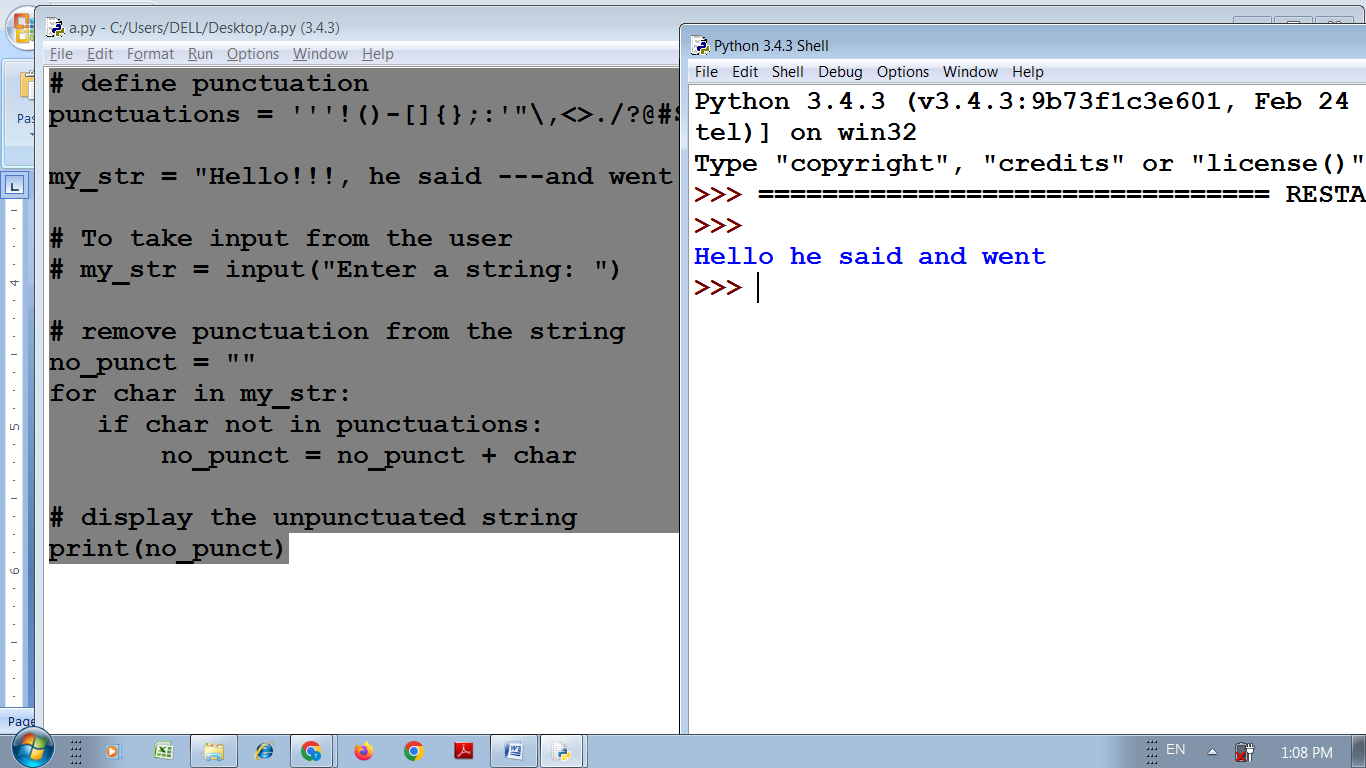
for char in my\_str:

if char not in punctuations:

no\_punct = no\_punct + char

# display the unpunctuated string

print(no\_punct)



# define punctuation

punctuations = '''!()-[]{};:'"\,<>./?@#$%^&\*\_~'''

# To take input from the user

my\_str = input("Enter a string: ")

# remove punctuation from the string

no\_punct = ""

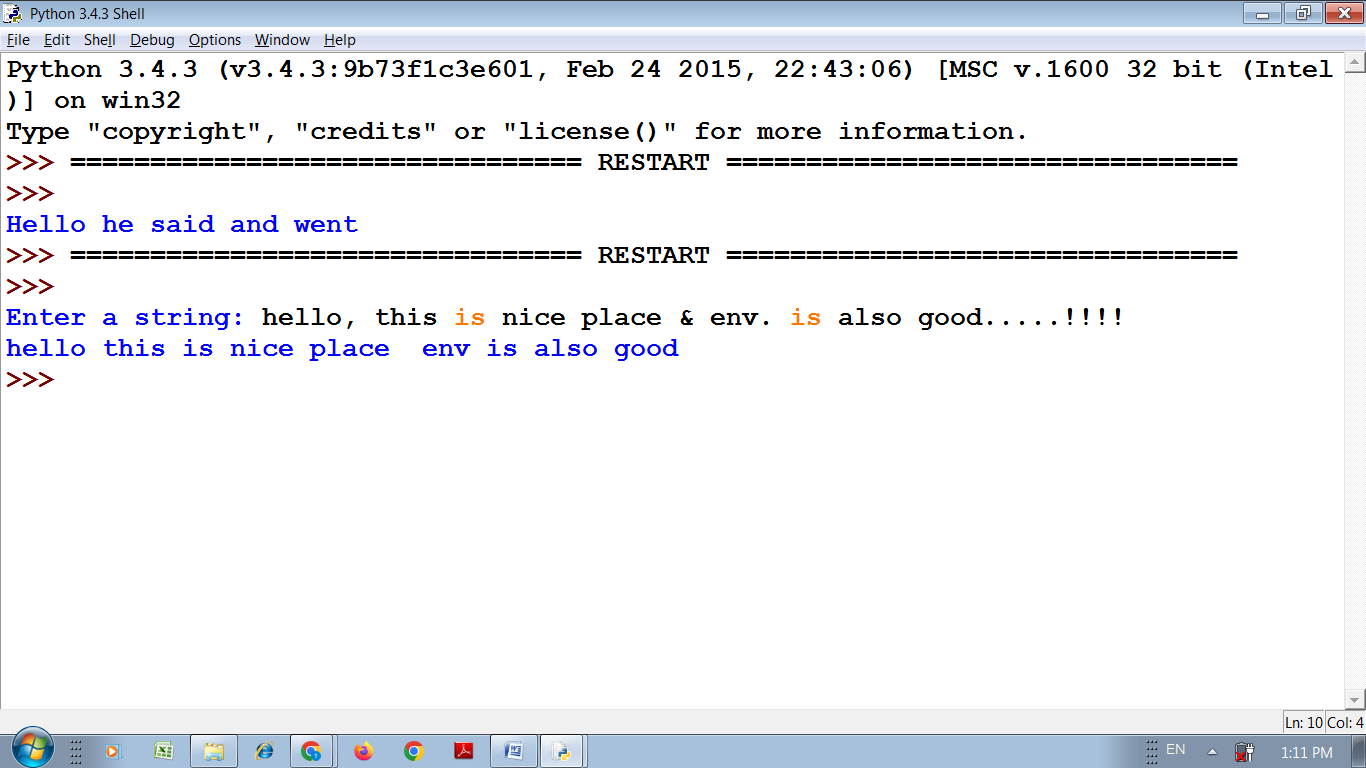
for char in my\_str:

if char not in punctuations:

no\_punct = no\_punct + char

# display the unpunctuated string

print(no\_punct)



Practical No :- 10

Aim :- Python program to demonstrate string functions.

Program :-

# Python3 program to show the

# working of upper() function

text = 'This is SYBCA Class'

# upper() function to convert

# string to upper case

print("\nConverted String:")

print(text.upper())

# lower() function to convert

# string to lower case

print("\nConverted String:")

print(text.lower())

# converts the first character to

# upper case and rest to lower case

print("\nConverted String:")

print(text.title())

#swaps the case of all characters in the string

# upper case character to lowercase and viceversa

print("\nConverted String:")

print(text.swapcase())

# convert the first character of a string to uppercase

print("\nConverted String:")

print(text.capitalize())

# original string never changes

print("\nOriginal String")

print(text)

Output :-

