1. Work with built-in packages

a. math module

#help("modules")

import math

print("the value of pi is:",math.pi)

import math as m

print("the value of pi is:",m.pi)

from math import pi,sqrt

print("the square root of 36 is:",math.sqrt(36))

print("the value of pi is:",math.pi)

print(math.cos(90))

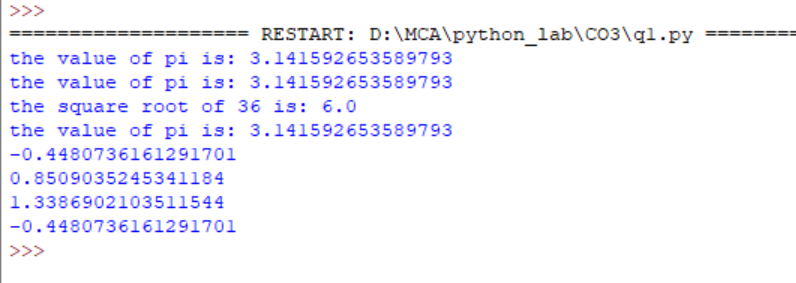
print(math.sin(45))

print(math.tan(180))

print(math.cos(90))

#print(math.cosec(90))

Output:



b.time module

import time

print("current time in sec:",time.time())

print("current time:",time.ctime())

print("current time after 30 sec:",time.ctime(time.time()+30))

t=time.localtime()

print("time t:",t)

print("current year:",t.tm\_year)

print("current month:",t.tm\_mon)

print("current day:",t.tm\_mday)

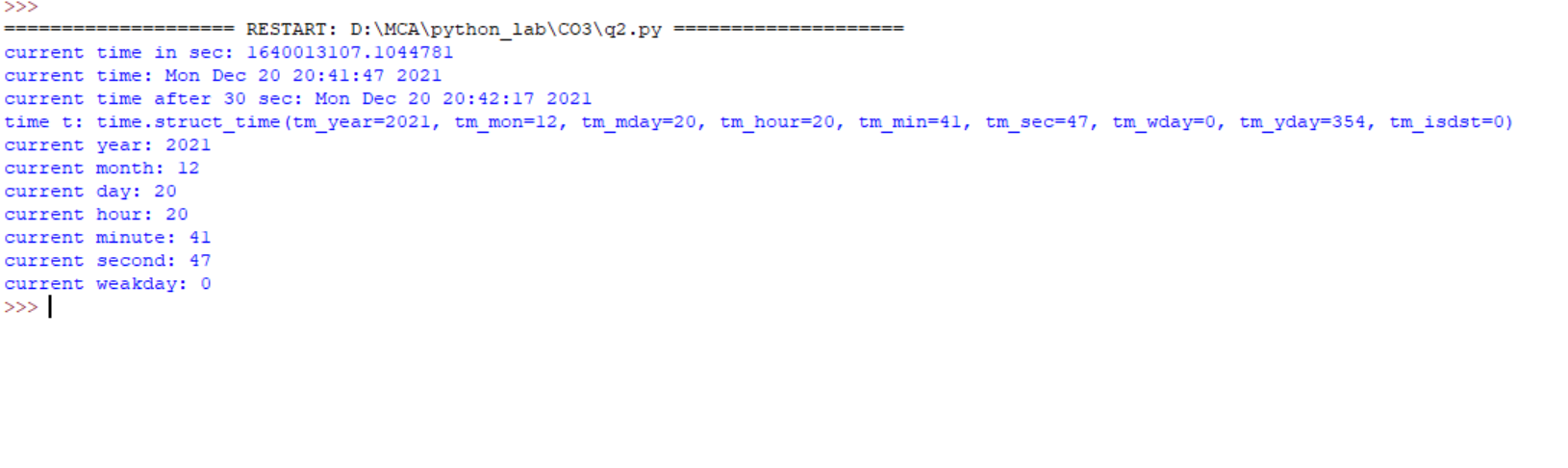
print("current hour:",t.tm\_hour)

print("current minute:",t.tm\_min)

print("current second:",t.tm\_sec)

print("current weakday:",t.tm\_wday)

output:



c. calender module

import calendar

mm=int(input("Enter month:"))

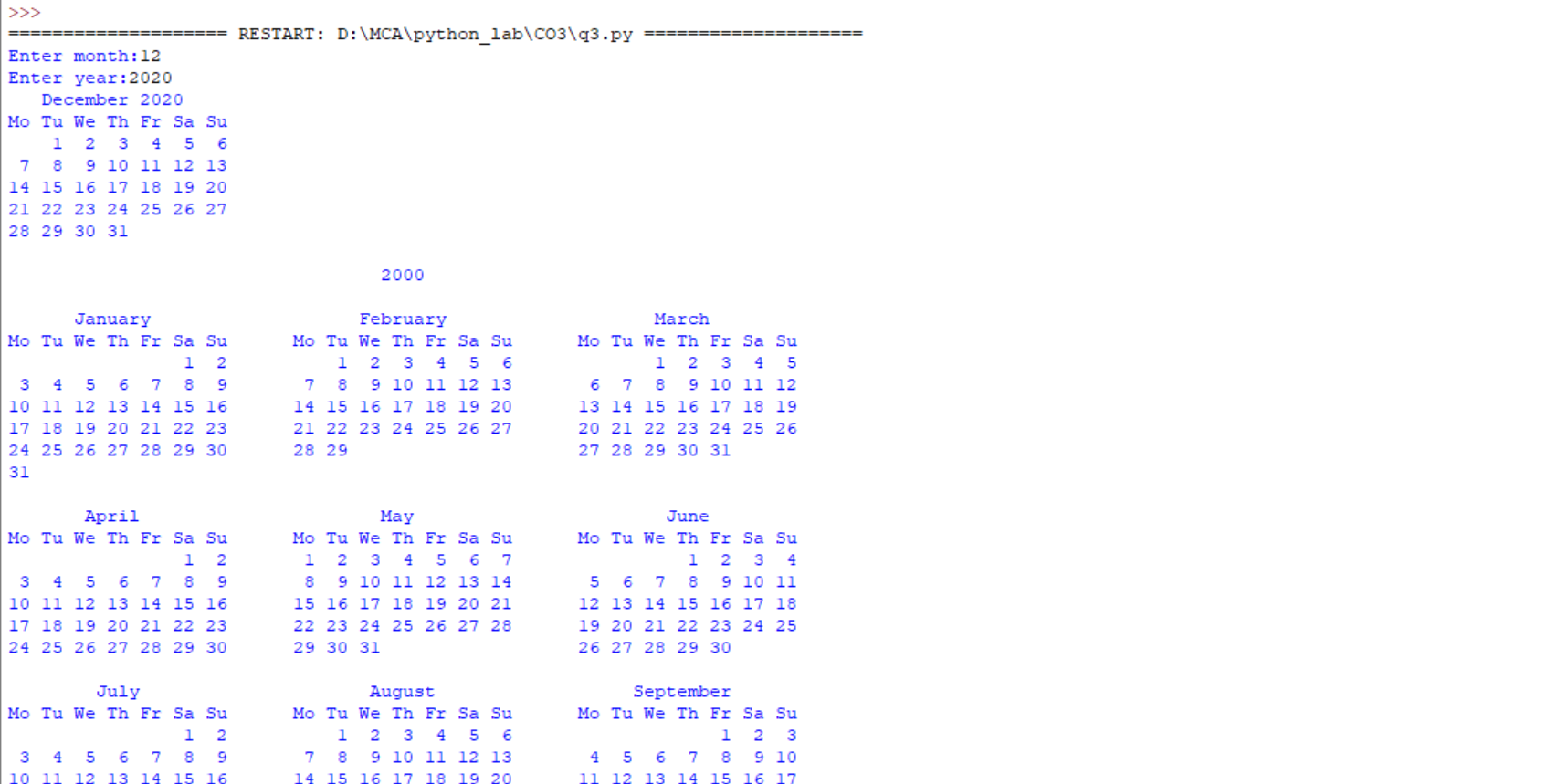
yy=int(input("Enter year:"))

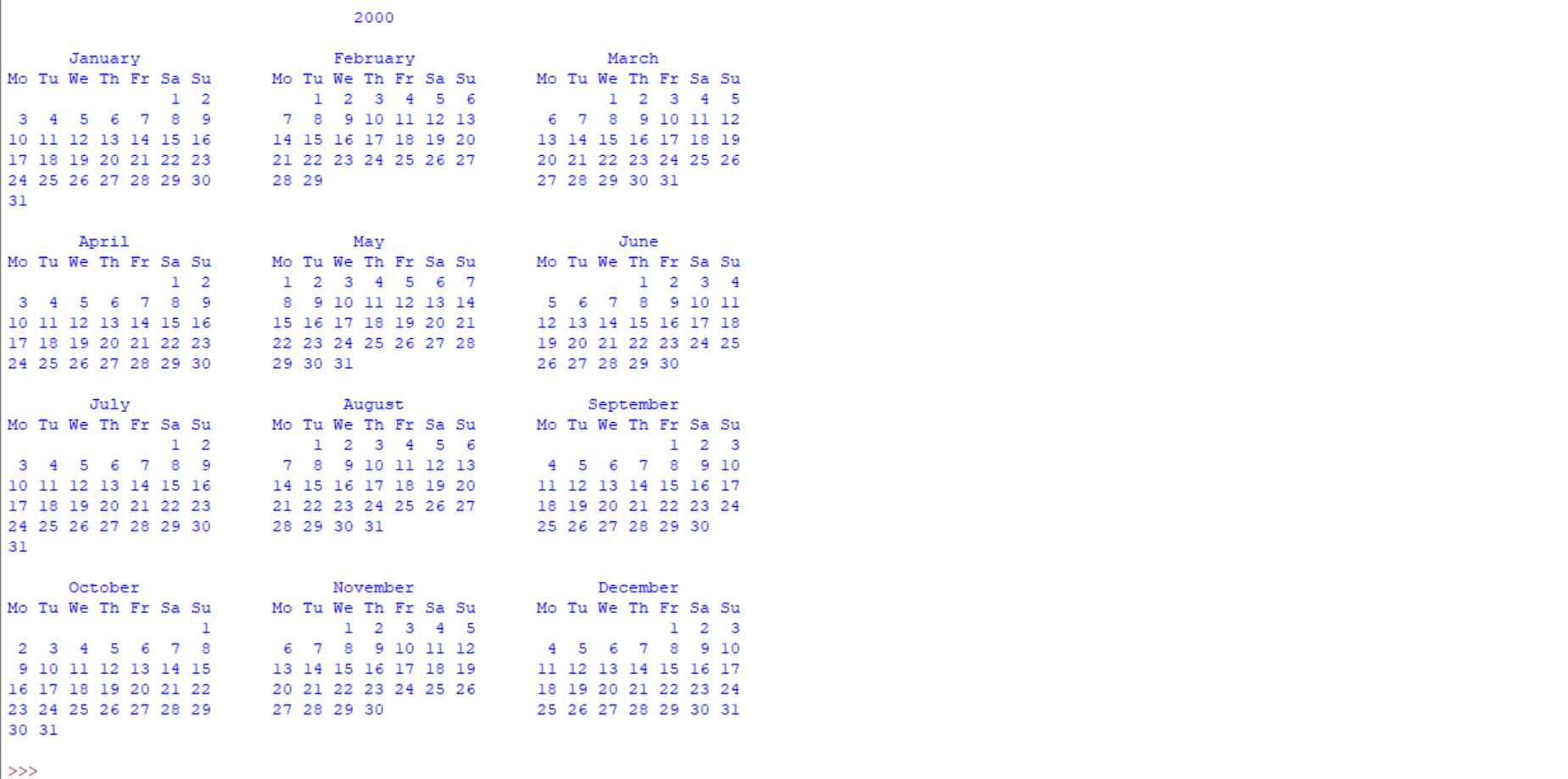
print(calendar.month(yy,mm)) #calendar of a given month

#print(calendar.calendar(2022)) #calendar of a given year

print(calendar.calendar(2000))

output:





d.datetime module

import datetime

t=datetime.time(22,56,44,5) #time class(hr,min,sec,microsec)

print(t)

print("Hour",t.hour)

print("Minute",t.minute)

print("Second",t.second)

print("Microsecond",t.microsecond)

print(".............................")

d=datetime.date.today() #date class

print(d)

td=datetime.timedelta(days=2) #timedelta class

print(td)

d2=d+td #adding 2 da

print(d2)

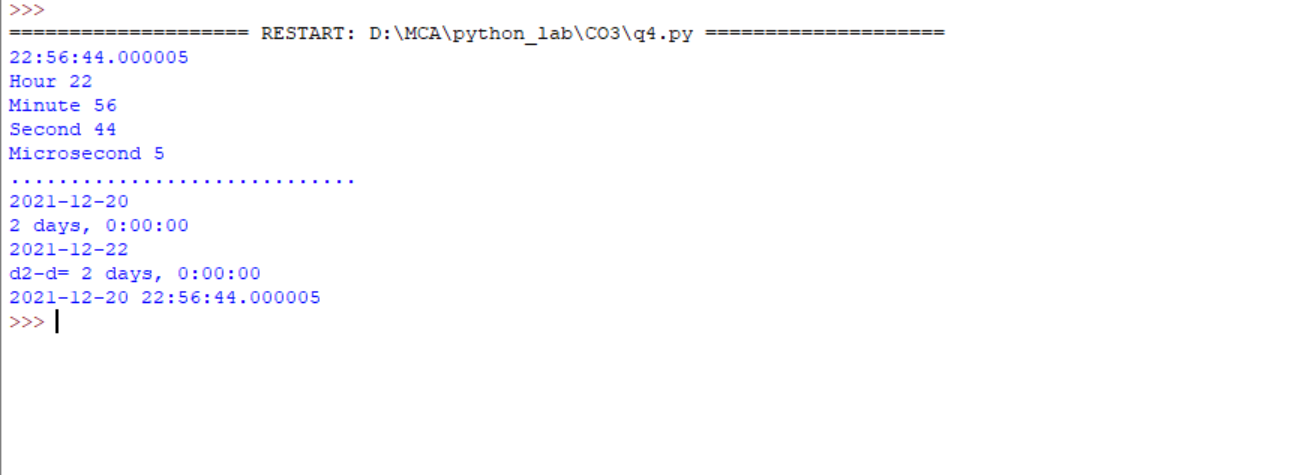
print("d2-d=",d2-d)

#d1=datetime.date.today()

dt=datetime.datetime.combine(d,t)

print(dt)

output:

e. import random

mylist = ["apple", "banana", "cherry"]

print(random.choice(mylist)) #Returns a random element from the given sequence

print(random.choices(mylist, k=2))

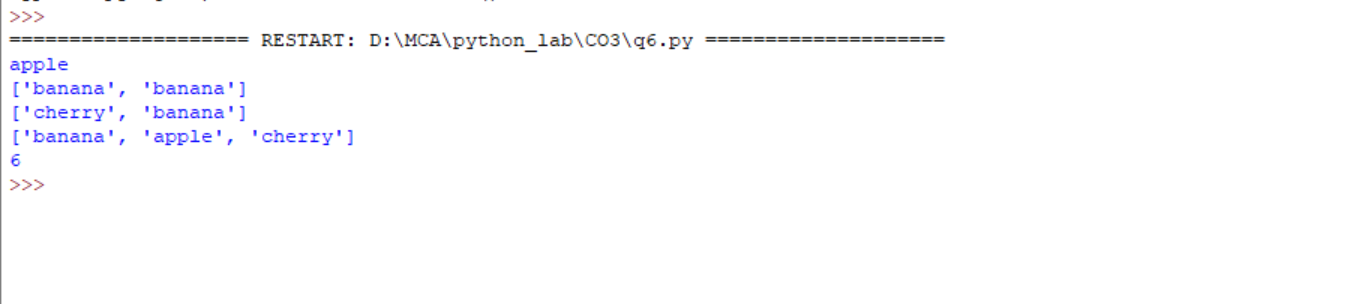
print(random.sample(mylist, k=2)) #Return a list that contains any 2 of the items from a list:

random.shuffle(mylist)

print(mylist) #Takes a sequence and returns the sequence in a random order

print(random.randrange(3, 9)) #Return a number between 3 and 9:

output:



f.

import statistics

l1=[1,2,3,4,4]

print("mean",statistics.mean(l1))

print("median",statistics.median(l1))

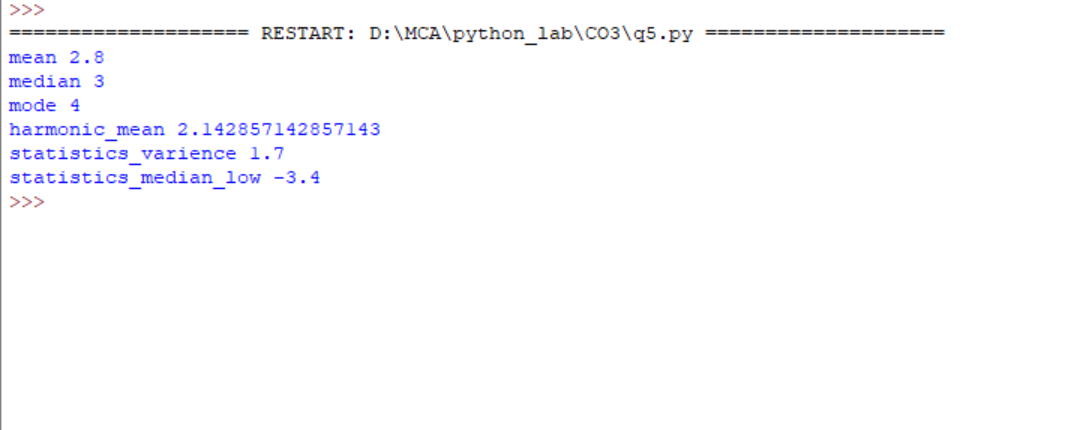
print("mode",statistics.mode(l1))

print("harmonic\_mean",statistics.harmonic\_mean(l1))

print("statistics\_varience",statistics.variance(l1))

print("statistics\_median\_low",statistics.median\_low([-11, 5.5, -3.4, 7.1, -9, 22]))

output:



2. Create a package graphics with modules rectangle, circle and sub-package 3D-graphics with modules cuboid and sphere. Include methods to find area and perimeter of respective figures in each module. Write programs that finds area and perimeter of figures by different importing statements. (Include selective import of modules and import \* statements)

Circle.py

pi=3.14

def area(pi,r):

return pi\*r\*r

def perimeter(pi,r):

return 2\*pi\*r

init.py

\_\_init\_\_

Rectangle.py

def area(a,b):

return a\*b

def perimeter(a,b):

return 2\*(a+b)

graphicsuse.py

from graphics import rectangle

from graphics import circle

print("Read values: \n Rectangle:\n")

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

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

r\_area=rectangle.area(l,b)

r\_perimeter=rectangle.perimeter(l,b)

print("Circle:\n")

r=int(input("Enter radius"))

area=circle.area(circle.pi,r)

perimeter=circle.perimeter(circle.pi,r)

print("Area of Rectangle:",r\_area)

print("Area of Circle:",area)

print("Perimeter of Rectangle:",r\_perimeter)

print("Perimeter of Circle:",perimeter)

output:

