**CYCLE-1**

**1)Write a program to demonstrate basic data type in python Hello, World! Program**

print("Hello, World!")

**OUTPUT**

Hello, World!

**2)Program to Print an Integer (Entered by the User)**

num1 = int(input('Enter the integer  number: '))

print('The number is {0} '.format(num1))

**OUTPUT**

Enter number: 5

The number is 5

**3)Program to Add Two Integers**

num1 = input('Enter first number: ')

num2 = input('Enter second number: ')

# Add two numbers

sum = int(num1)+int(num2)

# Display the sum

print("sum is",sum)

**OUTPUT**

Enter first number: 4

Enter second number: 5

sum is 9

**4)Program to Multiply Two Floating-Point Numbers**

num1 = input('Enter first number: ')

num2 = input('Enter second number: ')

# Add two numbers

sum = float(num1)\*float(num2)

# Display the sum

print("multiply is",sum)

**OUTPUT**

Enter first number: 4.5

Enter second number: 5

multiply is 22.5

**5)Program to Compute Quotient and Remainder**

def find(n, m):

    # for quotient

    q = n//m

    print("Quotient: ", q)

    # for remainder

    r = n%m

    print("Remainder", r)

# Driver Code

find(10, 3)

find(99, 5)

**OUTPUT**

Quotient: 3

Remainder 1

Quotient: 19

Remainder 4

**6)Program to Swap Two Numbers**

x = int(input('enter the value of x   :'))

y = int(input('enter the value of y  :'))

# create a temporary variable and swap the values

temp = x

x = y

y = temp

print("The value of x after swapping:",x)

print("The value of y after swapping:",y)

**OUTPUT**

enter the value of x :8

enter the value of y :9

The value of x after swapping: 9

The value of y after swapping: 8

**7)Program to Check Whether a Number is Even or Odd**

num = int(input("Enter a number: "))

if (num % 2) == 0:

   print(num," is even")

else:

   print(num," is odd")

**OUTPUT**

Enter a number: 5

5 is odd

**8)Program to Check Whether a Character is a Vowel or Consonant**

ch = input("Enter a character: ")

if(ch=='A' or ch=='a' or ch=='E' or ch =='e' or ch=='I'

 or ch=='i' or ch=='O' or ch=='o' or ch=='U' or ch=='u'):

    print(ch, "is a Vowel")

else:

    print(ch, "is a Consonant")

**OUTPUT**

Enter a character: t

t is a Consonant

**9)Program to Find the Largest Number Among Three Numbers**

x=int(input('enter the 1st number'))

y=int(input('enter the 2 nd number'))

z=int(input('enter the 3 rd number'))

if (x >= y) and (x >= z):

   largest = x

elif (y >= x) and (y >= z):

   largest = y

else:

   largest = z

   print("The largest number is", largest)

**OUTPUT**

enter the 1st number6

enter the 2 nd number7

enter the 3 rd number8

The largest number is 8

**10)Program to Check Leap Year**

year = int(input("enter the year"))

# To get year (integer input) from the user

# year = int(input("Enter a year: "))

if (year % 4) == 0:

   if (year % 100) == 0:

       if (year % 400) == 0:

           print("{0} is a leap year".format(year))

       else:

           print("{0} is not a leap year".format(year))

   else:

       print("{0} is a leap year".format(year))

else:

   print("{0} is not a leap year".format(year))

**OUTPUT**

enter the year2010

2010 is not a leap year

**11)Program to Check Whether a Number is Positive or Negative**

num = float(input("Enter a number: "))

if num >= 0:

   if num == 0:

       print("Zero")

   else:

       print("Positive number")

else:

   print("Negative number")

**OUTPUT**

Enter a number: 6

Positive number

**12)Program to Calculate the Sum of Natural Numbers**

num = int(input("Enter a number: "))

if num < 0:

   print("Enter a positive number")

else:

   sum = 0

   # use while loop to iterate un till zero

   while(num > 0):

       sum += num

       num -= 1

   print("The sum is",sum)

**OUTPUT**

Enter a number: 4

The sum is 10

**13)Program to Find Factorial of a Number**

num =int(input("enter a number"))

# To take input from the user

#num = int(input("Enter a number: "))

factorial = 1

# check if the number is negative, positive or zero

if num < 0:

   print("Sorry, factorial does not exist for negative numbers")

elif num == 0:

   print("The factorial of 0 is 1")

else:

   for i in range(1,num + 1):

       factorial = factorial\*i

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

**OUTPUT**

enter a number4

The factorial of 4 is 24

**14)Program to Generate Multiplication Table**

num = int(input("Display multiplication table of? "))

# Iterate 10 times from i = 1 to 10

for i in range(1, 11):

   print(num, 'x', i, '=', num\*i)

**OUTPUT**

Display multiplication table of? 5

5 x 1 = 5

5 x 2 = 10

5 x 3 = 15

5 x 4 = 20

5 x 5 = 25

5 x 6 = 30

5 x 7 = 35

5 x 8 = 40

5 x 9 = 45

5 x 10 = 50

**15)Program to Display Fibonacci Sequence**

nterms = int(input("How many terms? "))

# first two terms

n1, n2 = 0, 1

count = 0

# check if the number of terms is valid

if nterms <= 0:

   print("Please enter a positive integer")

elif nterms == 1:

   print("Fibonacci sequence upto",nterms,":")

   print(n1)

else:

   print("Fibonacci sequence:")

   while count < nterms:

       print(n1)

       nth = n1 + n2

       # update values

       n1 = n2

       n2 = nth

       count += 1

**OUTPUT**

How many terms? 5

Fibonacci sequence:

0

1

1

2

3

**16)Program to Find LCM of two Numbers**

def compute\_lcm(x, y):

   # choose the greater number

   if x > y:

       greater = x

   else:

       greater = y

   while(True):

       if((greater % x == 0) and (greater % y == 0)):

           lcm = greater

           break

       greater += 1

   return lcm

num1 = int(input(" 1 st number"))

num2 = int(input(" 2 st number"))

print("The L.C.M. is", compute\_lcm(num1, num2))

**OUTPUT**

1 st number3

2 st number5

The L.C.M. is 15

**17)Program to Count Number of Digits in an Integer**

# number of digits in a number

def countDigit(n):

    count = 0

    while n != 0:

        n //= 10

        count += 1

    return count

# Driver Code

n = int(input("enter the number"))

print("Number of digits : % d" % (countDigit(n)))

**OUTPUT**

enter the number875

Number of digits : 3

**18)Program to Reverse a Number**

Number = int(input("Please Enter any Number: "))

Reverse = 0.

while(Number > 0):

 Reminder = Number % 10

 Reverse = (Reverse \*10) + Reminder

 Number = Number //10

 print("\n Reverse of entered number is = %d" %Reverse)

**OUTPUT**

Please Enter any Number: 5567

Reverse of entered number is = 7

Reverse of entered number is = 76

Reverse of entered number is = 765

Reverse of entered number is = 7655

**19)Program to Calculate the Power of a Number**

base\_number = int(input("Enter the base number"))

exponent = int(input("Enter the exponent"))

power = base\_number \*\* exponent

print("Result is =",power)

**OUTPUT**

Enter the base number5

Enter the exponent4

Result is = 625

**20)Program to Check Whether a Number is Palindrome**

n=int(input("Enter number:"))

temp=n

rev=0

while(n>0):

  dig=n%10

  rev=rev\*10+dig

  n=n//10

if(temp==rev):

   print("The number is a palindrome!")

else:

  print("The number isn't a palindrome!")

**OUTPUT**

Enter number:444

The number is a palindrome!

**CYCLE 2**

1.Print your address

str1 = str(input("enter the address::"))

print(str1)

**OUTPUT**

enter the address::poothottathil

poothottathil

2.Display future leap years from current year to a final year entered by user.

s=int(input("enter starting year: "))

e=int(input("enter ending year: "))

print("leap years between",s,"and",e)

while (s<=e):

    if ((s%400==0) or ((s%4==0) and (s%100!=0))):

      print(s)

    s=s+1

**OUTPUT**

enter starting year: 2021

enter ending year: 2023

leap years between 2021 and 2023

3.List comprehensions: (a)Generate positive list of numbers from a given list of integers (b)Square of N numbers (c)Form a list of vowels selected from a given word (d)List ordinal value of each element of a word (Hint: use ord() to get ordinal values)

a) x=[]

n=int(input("enter list items : "))

for i in range(1,n+1):

    a=int(input("enter the value of %d item : "%i))

    x.append(a)

y=[num for num in x if num >= 0]

print("initial list of integers is :",x)

print("modified list of positive numbers is :",y)

**OUTPUT**

enter list items : 5

enter the value of 1 item : 3

enter the value of 2 item : 5

enter the value of 3 item : 7

enter the value of 4 item : 8

enter the value of 5 item : 3

initial list of integers is : [3, 5, 7, 8, 3]

modified list of positive numbers is : [3, 5, 7, 8, 3]

b)

x=[]

n=int(input("enter list items : "))

for i in range(1,n+1):

    a=int(input("enter the value of %d item : "%i))

    x.append(a)

b=(num\*\*2 for num in x)

print("initial list of N numbers :",x)

y=list(b)

print("square list of N numbers :",y)

**OUTPUT**

enter list items : 2

enter the value of 1 item : 4

enter the value of 2 item : 6

initial list of N numbers : [4, 6]

square list of N numbers : [16, 36]

c)

w=input("enter any word : ")

v=[l for l in w if l in 'aeiou' ]

print("list of vowels in",w," :",v)

**OUTPUT**

enter any word : anu

list of vowels in anu : ['a', 'u']

d)

w=input("enter any word : ")

for l in range(len(w)) :

  print("ASCII value is",w[l],"is",ord(w[l]))

**OUTPUT**

enter any word : anu

ASCII value is a is 97

ASCII value is n is 110

ASCII value is u is 117

4.Count the occurrences of each word in a line of text.

def word\_count(str):

    counts = dict()

    words = str.split()

    for word in words:

        if word in counts:

            counts[word] += 1

        else:

            counts[word] = 1

    return counts

print( word\_count('the quick brown fox jumps over the lazy dog.'))

**OUTPUT**

{'the': 2, 'quick': 1, 'brown': 1, 'fox': 1, 'jumps': 1, 'over': 1, 'lazy': 1, 'dog.': 1}

5.Prompt the user for a list of integers. For all values greater than 100, store ‘over’ instead.

x=[]

n=int(input("enter any integers :"))

for i in range(1,n+1):

  a=int(input())

  if (a>100):

    x.append('over')

  else:

    x.append(a)

print("modified list :" ,x)

enter any integers :4

20

50

400

100

modified list : [20, 50, 'over', 100]

6.Store a list of first names. Count the occurrences of ‘a’ within the list

list1 = ["a","b","a","a"]

print(list1.count("a"))

**OUTPUT**

3

7.Enter 2 lists of integers. Check (a) Whether list are of same length (b) whether list sums to same value (c) whether any value occur in both

list1 = [11, 22, 33, 44, 55]

list2 = [11, 22, 33, 44, 55,66]

s=len(list1)

m=len(list2)

print("length of list 1 is:",len(list1))

print("length of list 2 is:",len(list2))

if s==m:

 print("both list is same")

else:

 print("not same")

total1 = 0

ele = 0

while(ele < len(list1)):

    total1 = total1 + list1[ele]

    ele += 1

print("Sum of all elements in given list1 : ", total1)

total2 = 0

ele = 0

while(ele < len(list2)):

    total2 = total2 + list2[ele]

    ele += 1

print("Sum of all elements in given list2 : ", total2)

**OUTPUT**

length of list 1 is: 5

length of list 2 is: 6

not same

Sum of all elements in given list1 : 165

Sum of all elements in given list2 : 231

8.Get a string from an input string where all occurrences of first character replaced with ‘$’, except first character.[eg: onion -> oni$n]

Def change\_char(str1):

Char=str1[10]

Str1=str1.replace(char,’$’)

Str1=char+str1[1:]

Return str1

Print(change\_char(‘onion’))

**OUTPUT**

Oni$n

9.Create a string from given string where first and last characters exchanged. [eg: python -> nythop]

def change(str):

      return str[-1:] + str[1:-1] + str[:1]

str=input("enter any string : ")

print("modified string : ",change(str))

**OUTPUT**

enter any string : python

modified string : nythop

10.Accept the radius from user and find area of circle.

#program to find area of circle in Python using π

PI = 3.14

r = float(input("Enter the radius of the circle: "))

area = PI \* r \* r

print("%.2f" %area)

**OUTPUT**

Enter the radius of the circle: 3

28.26

11.Find biggest of 3 numbers entered.

#find biggest number

x=int(input('enter the 1st number'))

y=int(input('enter the 2 nd number'))

z=int(input('enter the 3 rd number'))

if (x > y) and (x > z):

   largest = x

elif (y > x) and (y > z):

   largest = y

else:

   largest = z

   print("The largest number is", largest)

**OUTPUT**

enter the 1st number7

enter the 2 nd number8

enter the 3 rd number9

The largest number is 9

12.Accept a file name from user and print extension of that.

# Python String Operations

str1 = str(input("enter the name"))

str2 =".php"

str3=(str1 + str2)# using \*

print(str3)

**OUTPUT**

enter the nameanu

anu.php

13.Create a list of colors from comma-separated color names entered by user. Display first and last colors.

color\_list = ["Red","Green","White" ,"Black"]

print(color\_list)

print( "%s %s"%(color\_list[0],color\_list[-1]))

**OUTPUT**

['Red', 'Green', 'White', 'Black']

Red Black

14.Accept an integer n and compute n+nn+nnn.

a = int(input("Input an integer : "))

n1 = int( "%s" % a )

print(n1)

n2 = int( "%s%s" % (a,a) )

print(n2)

n3 = int( "%s%s%s" % (a,a,a) )

print(n3)

print ("sum  =",n1+n2+n3)

**OUTPUT**

Input an integer : 1

1

11

111

sum = 123

15.Print out all colors from color-list1 not contained in color-list2.

list1 = ["White", "Black", "Red"]

list2 = ["Red", "Green"]

mylist = list1 + list2

mylist = list(dict.fromkeys(mylist))

print(mylist)

**OUTPUT**

['White', 'Black', 'Red', 'Green']

16.Create a single string separated with space from two strings by swapping the character at position 1.

def chars\_mix\_up(a, b):

  new\_a = b[:2] + a[2:]

  new\_b = a[:2] + b[2:]

  return new\_a + ' ' + new\_b

print(chars\_mix\_up('abc', 'xyz'))

**OUTPUT**

xyc abz

17.Sort dictionary in ascending and descending order.

import operator

d = {1: 2, 3: 4, 4: 3, 2: 1, 0: 0}

print('Original dictionary : ',d)

sorted\_d = sorted(d.items(), key=operator.itemgetter(1))

print('Dictionary in ascending order by value : ',sorted\_d)

sorted\_d = dict( sorted(d.items(), key=operator.itemgetter(1),reverse=True))

print('Dictionary in descending order by value : ',sorted\_d)

**OUTPUT**

Original dictionary : {1: 2, 3: 4, 4: 3, 2: 1, 0: 0}

Dictionary in ascending order by value : [(0, 0), (2, 1), (1, 2), (4, 3), (3, 4)]

Dictionary in descending order by value : {3: 4, 4: 3, 1: 2, 2: 1, 0: 0}

18.Merge two dictionaries.

def Merge(dict1, dict2):

    res = {\*\*dict1, \*\*dict2}

    return res

dict1 = {'a': 10, 'b': 8}

dict2 = {'d': 6, 'c': 4}

dict3 = Merge(dict1, dict2)

print(dict3)

**OUTPUT**

{'a': 10, 'b': 8, 'd': 6, 'c': 4}

19.Find gcd of 2 numbers.

import math

print("The gcd of 60 and 48 is : ", end="")

print(math.gcd(60, 48))

**OUTPUT**

The gcd of 60 and 48 is : 12

20.From a list of integers, create a list removing even num

list = [11, 22, 33, 44, 55]

print("list before removing EVENnumbers")

print (list)

for i  in list:

  if(i%2 == 0):

      list.remove(i)

print ("list after removing EVEN numbers:")

print (list)

**OUTPUT**

list before removing EVENnumbers

[11, 22, 33, 44, 55]

list after removing EVEN numbers:

[11, 33, 55]

**CYCLE 3**

1. Program to find the factorial of a number

2. Generate Fibonacci series of N terms

3. Find the sum of all items in a list

4. Generate a list of four digit numbers in a given range with all their digits even and the number is a perfect square.

5. Display the given pyramid with step number accepted from user. Eg: N=4 1 2 4 3 6 9 4 8 12 16

6. Count the number of characters (character frequency) in a string.

7. Add ‘ing’ at the end of a given string. If it already ends with ‘ing’, then add ‘ly’

8. Accept a list of words and return length of longest word.

9. Construct following pattern using nested loop \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

10. Generate all factors of a number.

11. Write lambda functions to find area of square, rectangle and triangle

1. Program to find the factorial of a number

num=int(input("enter the number"))

factorial = 1

# check if the number is negative, positive or zero

if num < 0:

   print("Sorry, factorial does not exist for negative numbers")

elif num == 0:

   print("The factorial of 0 is 1")

else:

   for i in range(1,num + 1):

       factorial = factorial\*i

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

**OUTPUT**

enter the number7

The factorial of 7 is 5040

2. Generate Fibonacci series of N terms

n1, n2 = 0, 1

count = 0

# check if the number of terms is valid

if nterms <= 0:

   print("Please enter a positive integer")

elif nterms == 1:

   print("Fibonacci sequence upto",nterms,":")

   print(n1)

else:

   print("Fibonacci sequence:")

   while count < nterms:

       print(n1)

       nth = n1 + n2

       # update values

       n1 = n2

       n2 = nth

       count += 1

**OUTPUT**

How many terms? 5

Fibonacci sequence:

0

1

1

2

3

3. Find the sum of all items in a list

# Python program to find sum of elements in list

total = 0

# creating a list

list1 = [11, 5, 17, 18, 23]

# Iterate each element in list

# and add them in variale total

for ele in range(0, len(list1)):

    total = total + list1[ele]

# printing total value

print("Sum of all elements in given list: ", total)

**OUTPUT**

Sum of all elements in given list: 74

4. Generate a list of four digit numbers in a given range with all their digits even and the number is a perfect square.

l=int(input("Enter lower range: "))

r=int(input("Enter upper range: "))

for i in range(l,r+1,1):

   for j in range(32,100,1):

      if i == j\*j:

         string = str(i)

         if int(string[0])%2 == 0 and int(string[1])%2 == 0 and int(string[2])%2 == 0 and int(string[3])%2 == 0:

           print(i)

**OUTPUT**

Enter lower range: 3

Enter upper range: 6

5. Display the given pyramid with step number accepted from user. Eg: N=4 1 2 4 3 6 9 4 8 12 16

row = int(input('Enter how many lines? '))

for i in range(1,row+1):

    for j in range(1, row+1-i):

        print(' ', end='')

    for j in range(1,i+1):

        print(j, end='')

    for j in range(i-1,0,-1):

        print(j, end='')

    print()

**OUTPUT**

Enter how many lines? 3

1

121

12321

6. Count the number of characters (character frequency) in a string.

row = int(input('Enter how many lines? '))

# Generating pattern

for i in range(1,row+1):

    # for space

    for j in range(1, row+1-i):

        print(' ', end='')

    # for increasing pattern

    for j in range(1,i+1):

        print(j, end='')

    # for decreasing pattern

    for j in range(i-1,0,-1):

        print(j, end='')

    # Moving to next line

    print()

**OUTPUT**

{'g': 2, 'o': 3, 'l': 1, 'e': 1, '.': 1, 'c': 1, 'm': 1}

7. Add ‘ing’ at the end of a given string. If it already ends with ‘ing’, then add ‘ly’

def add\_string(str1):

  length = len(str1)

  if length > 2:

    if str1[-3:] == 'ing':

      str1 += 'ly'

    else:

      str1 += 'ing'

  return str1

print(add\_string('ab'))

print(add\_string('abc'))

print(add\_string('string'))

**OUTPUT**

ab

abcing

stringly

8. Accept a list of words and return length of longest word.

def longestLength(a):

    max1 = len(a[0])

    temp = a[0]

    # for loop to traverse the list

    for i in a:

        if(len(i) > max1):

            max1 = len(i)

            temp = i

    print("The word with the longest length is:", temp,

          " and length is ", max1)

# Driver Program

a = input("Enter a list elements separated by space ")

a = a.split()

longestLength(a)

**OUTPUT**

Enter a list elements separated by space qas dewf

The word with the longest length is: dewf and length is 4

9. Construct following pattern using nested loop \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

print("Print equilateral triangle Pyramid using stars ")

size = 7

m = (2 \* size) - 2

for i in range(0, size):

    for j in range(0, m):

        print(end=" ")

    m = m - 1  # decrementing m after each loop

    for j in range(0, i + 1):

        # printing full Triangle pyramid using stars

        print("\* ", end=' ')

    print(" ")

**OUTPUT**

Print equilateral triangle Pyramid using stars

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

\* \* \* \* \* \*

\* \* \* \* \* \* \*

10. Generate all factors of a number.

def print\_factors(x):

   print("The factors of",x,"are:")

   for i in range(1, x + 1):

       if x % i == 0:

           print(i)

num = 8

print\_factors(num)

**OUTPUT**

The factors of 8 are:

1

2

4

8

11. Write lambda functions to find area of square, rectangle and triangle

import math

t\_peri =lambda p,q,r :p+q+r

r\_area=lambda len,ht :len\*ht

c\_peri=lambda rad :2\*math.pi\*rad

c\_area=lambda rad :math.pi\*rad\*rad

print("perimeter of triangle (10,20,15) is ",t\_peri(10,20,15))

print("area of rectangle (30,20) is",r\_area(30,20))

print("perimeter of circle (10) is:",c\_peri(10))

print("area of circle (15) is: ",c\_area(15))

**OUTPUT**

perimeter of triangle (10,20,15) is 45

area of rectangle (30,20) is 600

perimeter of circle (10) is: 62.83185307179586

area of circle (15) is: 706.8583470577034

**CYCLE 4**

1A) Write a python program to create a package (college),sub-package (alldept),modules(it,cse) and create admin and cabin function to module?

Main.py

import COLLEGE.format  
import IT  
import CSE  
  
print(COLLEGE.format.coll())  
  
print(IT.details())  
print(CSE.time())

format.py

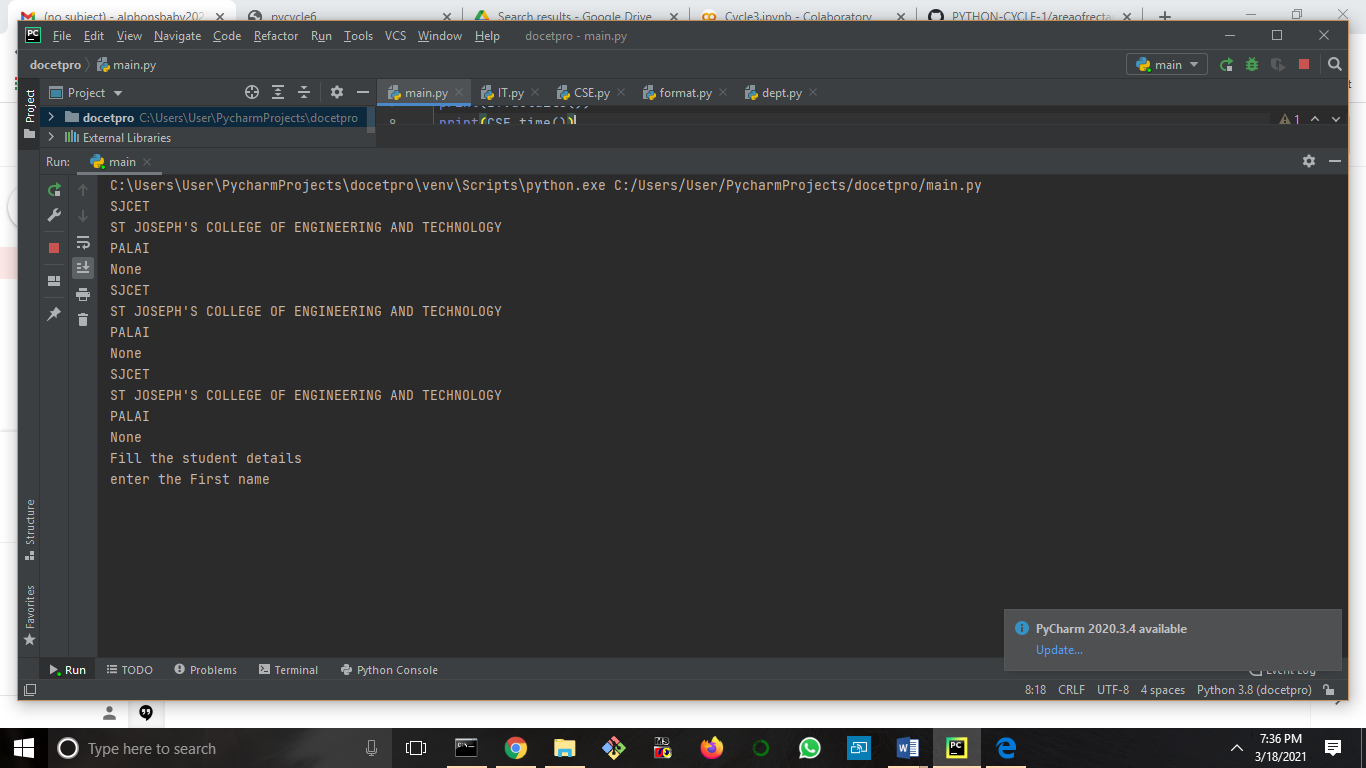
def coll():  
 print("SJCET")  
 print("ST JOSEPH'S COLLEGE OF ENGINEERING AND TECHNOLOGY")  
 print("PALAI")

it.py

import COLLEGE.format  
print(COLLEGE.format.coll())  
def details():  
 print("Fill the student details")  
 str1 = str(input("enter the First name"))  
 str3 = stra(input("enter the last name"))  
 print(str1)  
 print(str3)

cse.py

import COLLEGE.format  
  
print(COLLEGE.format.coll())  
  
def time():  
 num = int(input("enter the age"))  
 print(num)

**OUTPUT** 

2B) Write a python program to create a package(Engg), sub-package( years),modules (sem) and create staff and student function to module?

Main.py

import engg.be  
import sem  
print(engg.be.type())  
print(sem.staff())  
print(sem.stud())

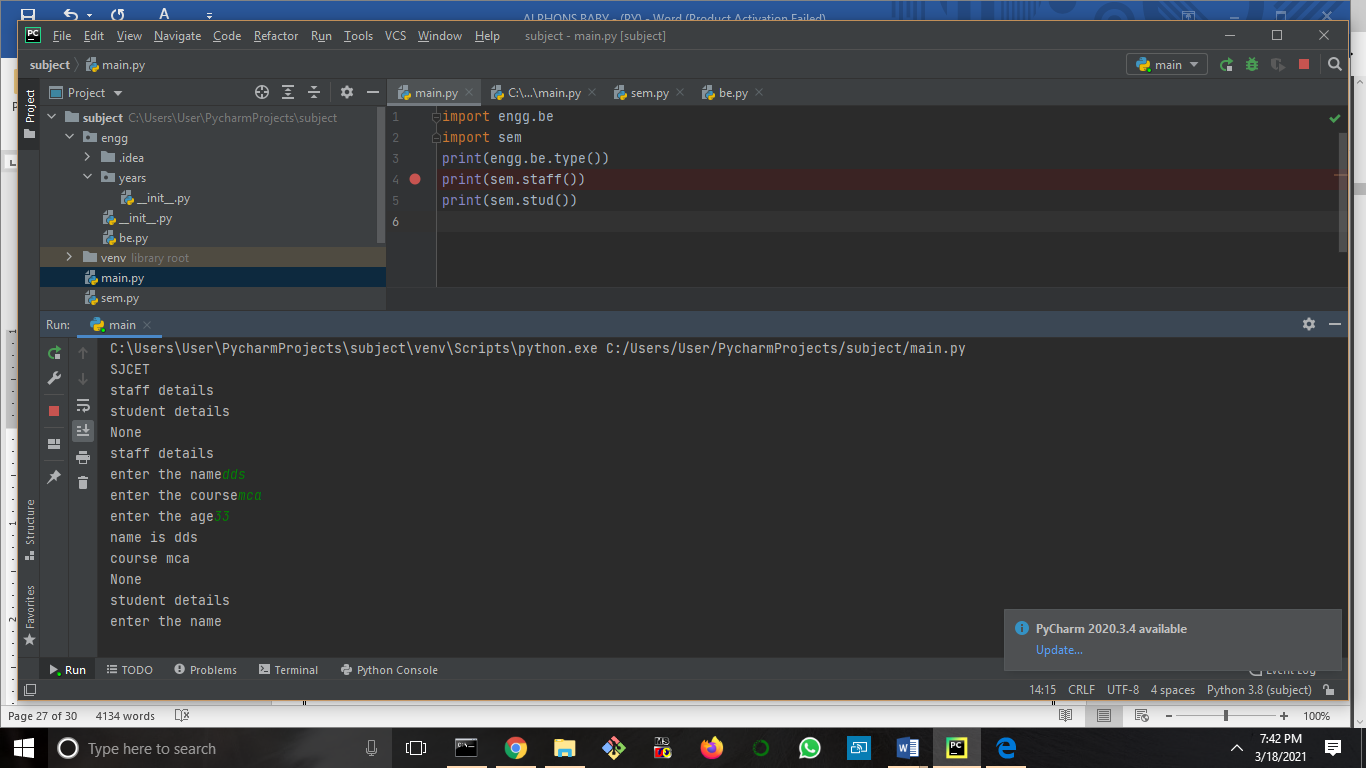
sem.py

def staff():  
 print("staff details")  
 str3=str(input("enter the name"))  
 str4 =str(input("enter the course"))  
 a = int(input("enter the age"))  
 print("name is",str3)  
 print("course",str4)  
  
def stud():  
 print("student details")  
 str6 = str(input("enter the name"))  
 str7 = str(input("enter the course"))  
 print("name is",str6)  
 print("course",str7)

be.py

def type():  
 print("SJCET")  
 print("staff details")  
 print("student details")

output



3, 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

Main.py

import graphics.circule  
import format  
import graphics.rectangle  
import graphics.subgraphics.sphere  
import graphics.subgraphics.cuboid  
print(format.form())  
print(graphics.circule.cirarea())  
print(graphics.rectangle.rect())  
print(graphics.rectangle.rect1())  
print(graphics.subgraphics.cuboid.cub())  
print(graphics.subgraphics.sphere.sph())

format.py

def form():  
 print("AREA OF CIRCULE,RECTANGLE,CUBOID,SPHERE")

circule.py

def cirarea():  
 print("AREA OF CIRCULE")  
 pi=3.73  
 rad = float(input('Please Enter the value '))  
 a= pi\* rad \*rad  
 p= 2\*pi \*rad  
 print("area of circle is: ", a)  
 print("perimeter of circle is:", p)

rectangle.py

def rect():  
 print("AREA OF RECTANGLE")  
 p = float(input('Please Enter the value '))  
 q = float(input('Please Enter the value '))  
 r = float(input('Please Enter the value '))  
 t\_peri=p+q+r  
 print("perimeter of triangle (10,20,15) is ",t\_peri)  
def rect1():  
 length = float(input('Please Enter the Length : '))  
 height = float(input('Please Enter the Height: '))  
 r\_area=length\*height  
 print("area of rectangle (30,20) is",r\_area)

cuboid.py

def cud():  
 print("AREA OF CUBOID")  
 length = int(input('Please Enter the Length of a Cuboid: '))  
 width = int(input('Please Enter the Width of a Cuboid: '))  
 height = int(input('Please Enter the Height of a Cuboid: '))  
 SA = 2 \* (length \* width + length \* height + width \* height)  
 Volume = length \* width \* height  
 print("\n The Surface Area of a Cuboid = %.2f " %SA)  
 print(" The Volume of a Cuboid = %.2f" %Volume);

sphere.py

def sph():  
 print("AREA OF SPHERE")  
 pi=22/7  
 radian = float(input('Radius of sphere: '))  
 sur\_area = 4 \* pi \* radian \*\*2  
 volume = (4/3) \* (pi \* radian \*\* 3)  
 print("Surface Area is: ", sur\_area)  
 print("Volume is: ", volume)

**OUTPUT**

C:\Users\User\PycharmProjects\area\venv\Scripts\python.exe C:/Users/User/PycharmProjects/area/main.py

AREA OF CIRCULE,RECTANGLE,CUBOID,SPHERE

None

AREA OF CIRCULE

Please Enter the value 3

area of circle is: 33.57

perimeter of circle is: 22.38

None

AREA OF RECTANGLE

Please Enter the value 4

Please Enter the value 5

Please Enter the value 6

perimeter of triangle (10,20,15) is 15.0

None

Please Enter the Length : 2

Please Enter the Height: 3

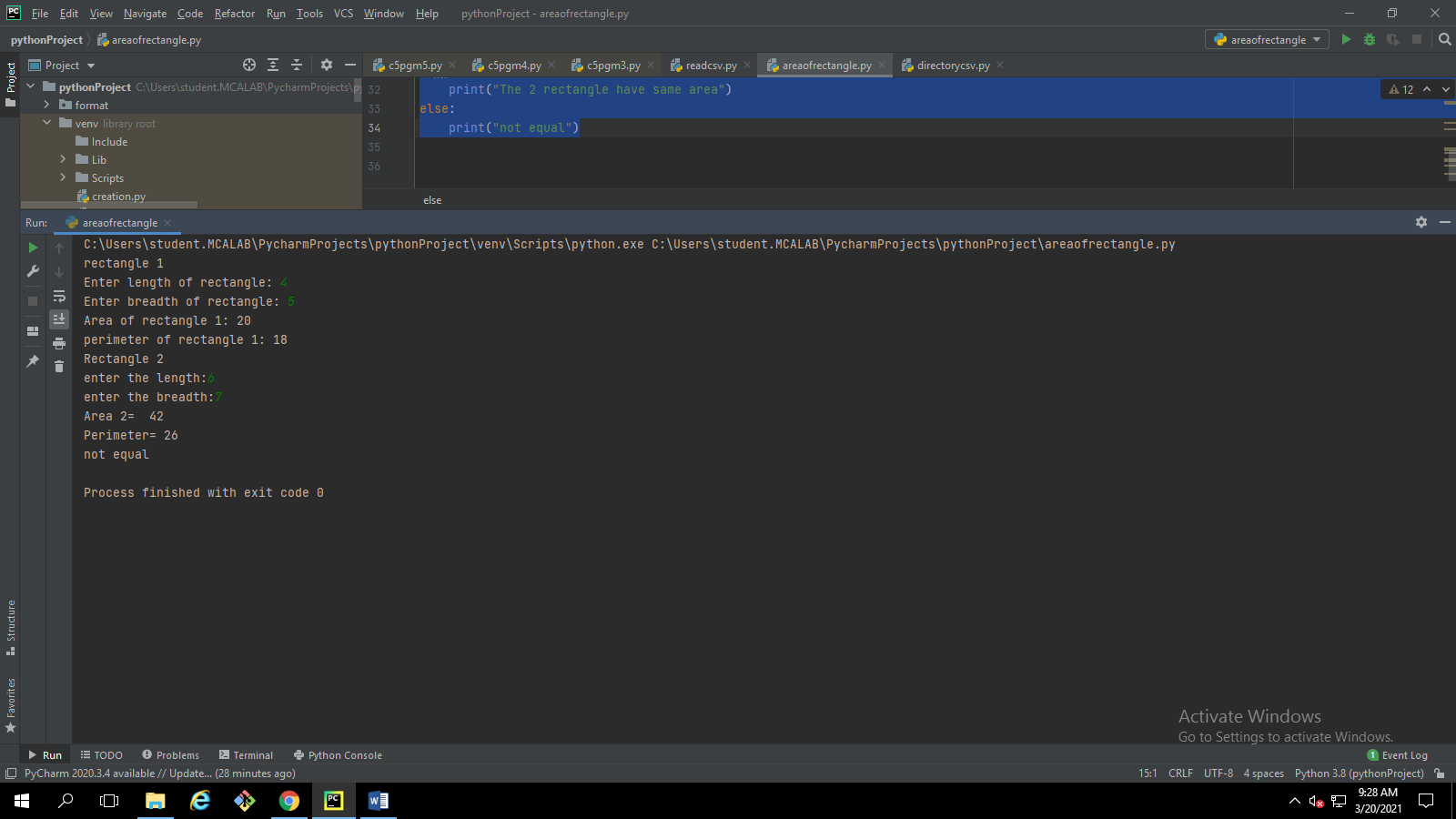
area of rectangle (30,20) is 6.0

None

**CYCLE 5**

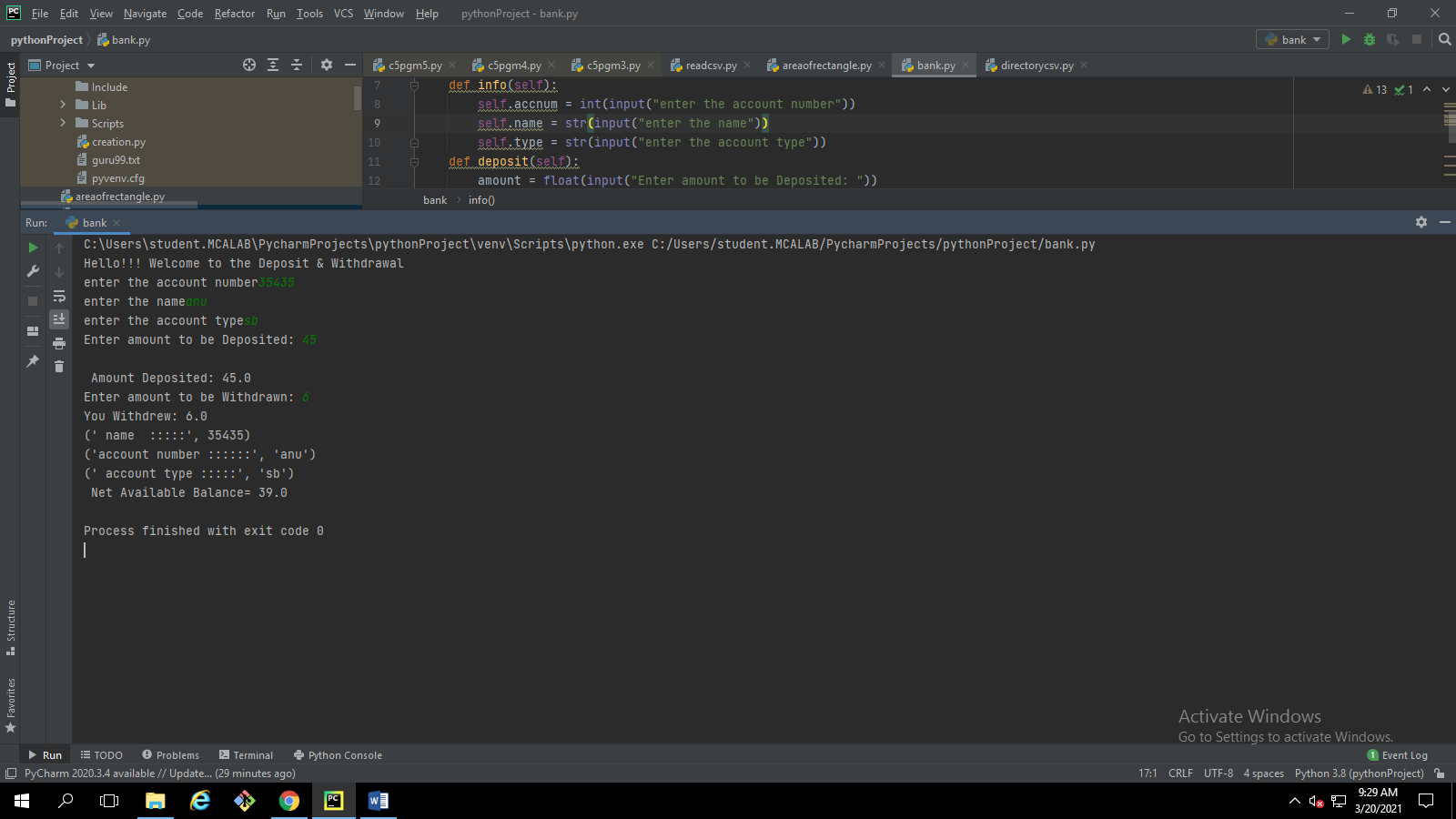
1. Create  Rectangle  class  with  attributes  length  and  breadth  and  methods  to  find  area  and  
perimeter. Compare two Rectangle objects by their area.

#1. Create Rectangle class with attributes length and breadth and methods to find area and  
#perimeter. Compare two Rectangle objects by their area.  
  
class Rectangle:  
 def \_\_init\_\_(self, length, breadth):  
 self.length = length  
 self.breadth = breadth  
  
 def area(self):  
 return self.length \* self.breadth  
 def peri(self):  
 return 2\*(self.length + self.breadth)  
  
print("rectangle 1")  
a = int(input("Enter length of rectangle: "))  
b = int(input("Enter breadth of rectangle: "))  
obj = Rectangle(a, b)  
print("Area of rectangle 1:", obj.area())  
print("perimeter of rectangle 1:", obj.peri())  
  
  
print("Rectangle 2")  
a=int(input("enter the length:"))  
b= int(input("enter the breadth:"))  
ob=Rectangle(a,b)  
  
  
print("Area 2= ",ob.area())  
print("Perimeter=",ob.peri())  
  
if obj.area() == ob.area():  
 print("The 2 rectangle have same area")  
else:  
 print("not equal")

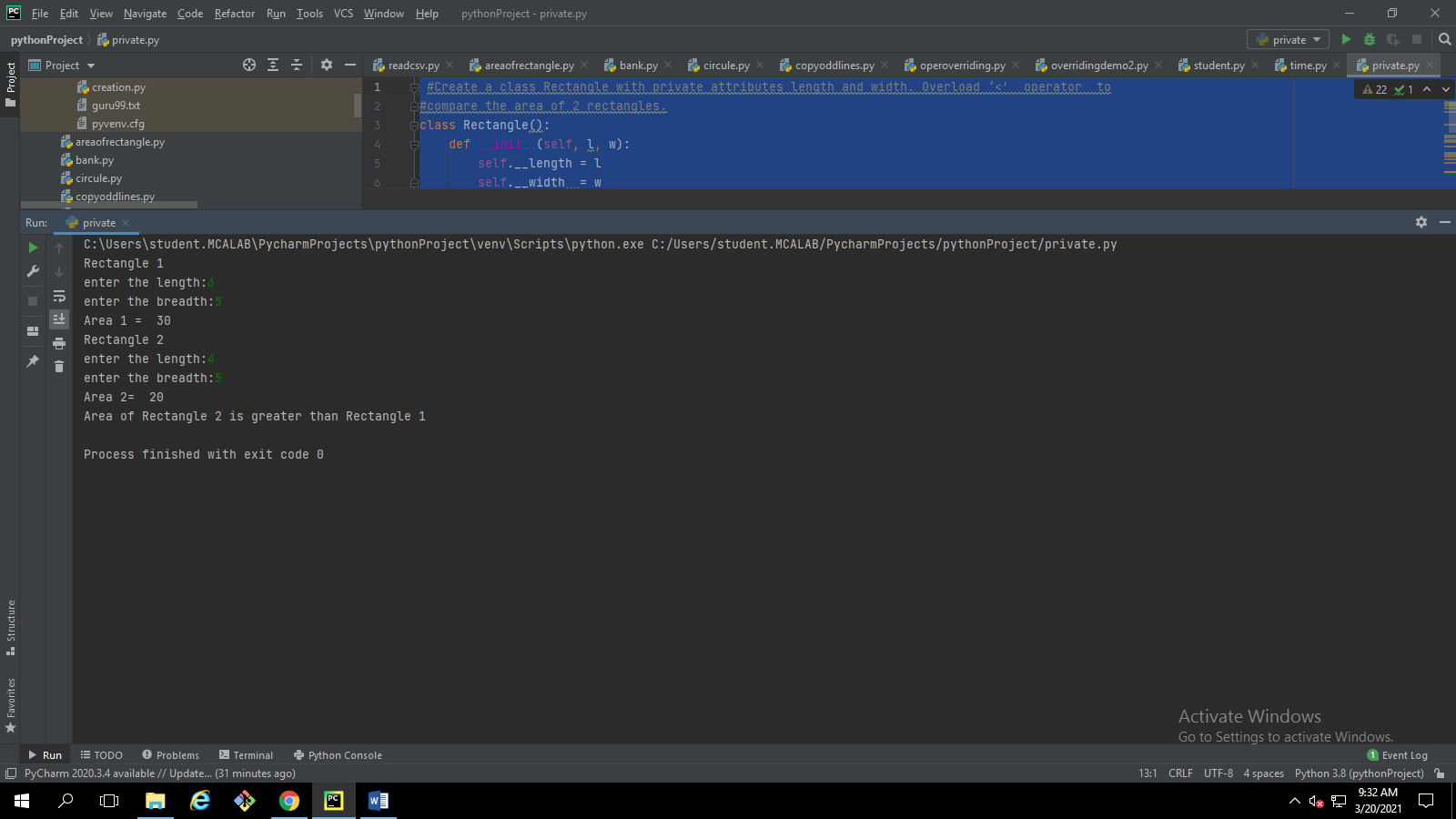
**OUTPUT** 

2.  Create a Bank account with members account number, name, type of account and balance.    
Write constructor and methods to deposit at the bank and withdraw an amount from the bank.

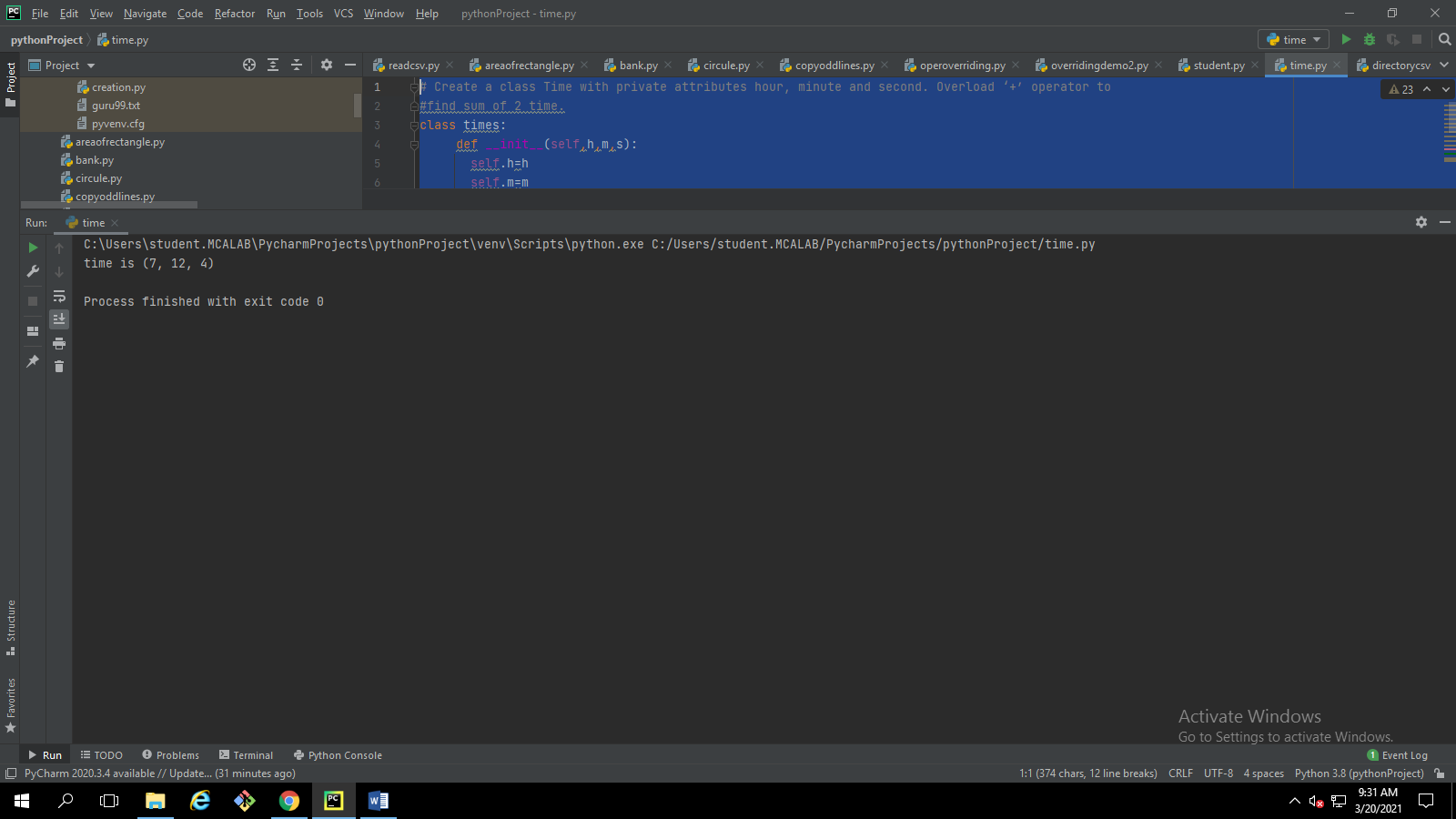
# Create a Bank account with members account number, name, type of account and balance.  
#Write constructor and methods to deposit at the bank and withdraw an amount from the bank.  
class bank:  
 def \_\_init\_\_(self):  
 self.balance=0  
 print("Hello!!! Welcome to the Deposit & Withdrawal ")  
 def info(self):  
 self.accnum = int(input("enter the account number"))  
 self.name = str(input("enter the name"))  
 self.type = str(input("enter the account type"))  
 def deposit(self):  
 amount = float(input("Enter amount to be Deposited: "))  
 self.balance += amount  
 print("\n Amount Deposited:", amount)  
  
 def withdraw(self):  
 amount = float(input("Enter amount to be Withdrawn: "))  
 if self.balance >= amount:  
 self.balance -= amount  
 print("You Withdrew:", amount)  
 else:  
 print("Insufficient balance ")  
  
 def display(self):  
 print((" name :::::",self.accnum ))  
 print(("account number ::::::", self.name,))  
 print((" account type :::::", self.type))  
 print(" Net Available Balance=", self.balance)  
s =bank()  
s.info()  
s.deposit()  
s.withdraw()  
s.display()

**OUTPUT**   
3. Create a class Rectangle with private attributes length and width. Overload ‘<’  operator  to  
compare the area of 2 rectangles.

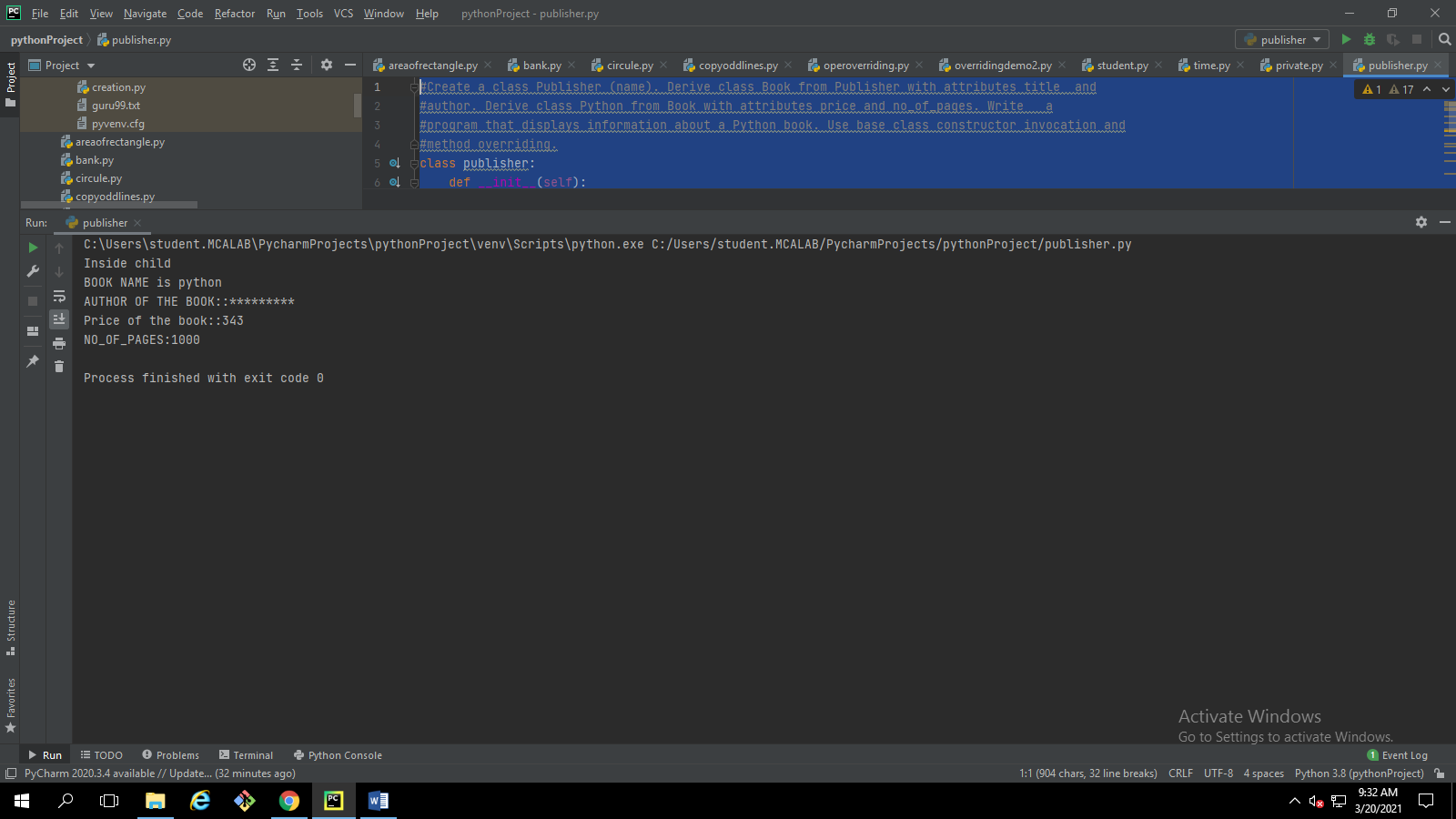
#Create a class Rectangle with private attributes length and width. Overload ‘<’ operator to  
#compare the area of 2 rectangles.  
class Rectangle():  
 def \_\_init\_\_(self, l, w):  
 self.\_\_length = l  
 self.\_\_width = w  
  
 def area(self):  
 return self.\_\_length\*self.\_\_width  
  
 def \_\_lt\_\_(self):  
 if (rec1 < rec2):  
 print("Area of Rectangle 1 is less than Rectangle 2")  
 else:  
 print("Area of Rectangle 2 is greater than Rectangle 1")  
  
print("Rectangle 1")  
a=int(input("enter the length:"))  
b= int(input("enter the breadth:"))  
obj=Rectangle(a,b)  
  
print("Area 1 = ",obj.area())  
  
print("Rectangle 2")  
a=int(input("enter the length:"))  
b= int(input("enter the breadth:"))  
ob=Rectangle(a,b)  
print("Area 2= ",ob.area())  
  
#compairing  
  
rec1 = obj.area()  
rec2 = ob.area()  
obje=Rectangle(rec1,rec2)  
obje.\_\_lt\_\_()

**OUTPUT**   
4. Create a class Time with private attributes hour, minute and second. Overload ‘+’ operator to  
find sum of 2 time.

# Create a class Time with private attributes hour, minute and second. Overload ‘+’ operator to  
#find sum of 2 time.  
class times:  
 def \_\_init\_\_(self,h,m,s):  
 self.h=h  
 self.m=m  
 self.s=s  
 def \_\_add\_\_(self, other):  
 return self.h + other.h, self.m + other.m, self.s + other.s  
v1 = times(2,10,3)  
v2 = times(5,2,1)  
v3 = v1+v2  
print("time is",v3)

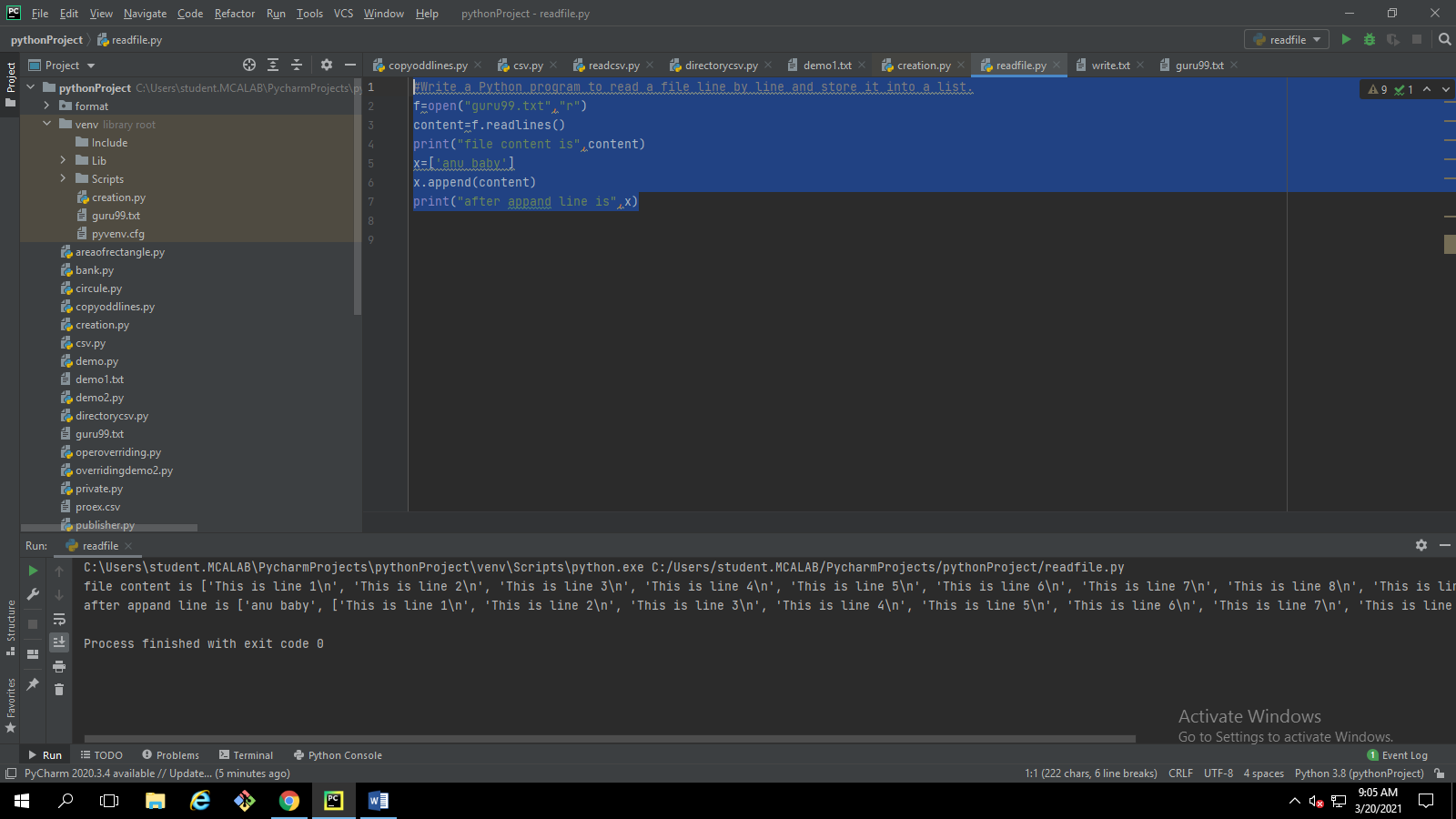
**OUTPUT**   
5. Create a class Publisher (name). Derive class Book from Publisher with attributes title  and  
author. Derive class Python from Book with attributes price and no\_of\_pages. Write   a  
program that displays information about a Python book. Use base class constructor invocation and  
method overriding.

#Create a class Publisher (name). Derive class Book from Publisher with attributes title and  
#author. Derive class Python from Book with attributes price and no\_of\_pages. Write a  
#program that displays information about a Python book. Use base class constructor invocation and  
#method overriding.  
class publisher:  
 def \_\_init\_\_(self):  
 self.value = "Inside Parent"  
 def title(self):  
 print(self.value)  
 print("TITLE OF BOOK")  
 def author(self):  
 print("AUTHOR OF THE BOOK::\*\*\*\*\*\*\*\*\*")  
  
class book(publisher):  
 def \_\_init\_\_(self):  
 self.value = "Inside child"  
 def title(self):  
 print(self.value)  
 print("BOOK NAME is python")  
  
class python(book):  
 def price(self):  
 print("Price of the book::343")  
  
 def no\_of\_page(self):  
 print("NO\_OF\_PAGES:1000")  
  
  
obj=python()  
obj.title()  
obj.author()  
obj.price()  
obj.no\_of\_page()

**OUTPUT** 

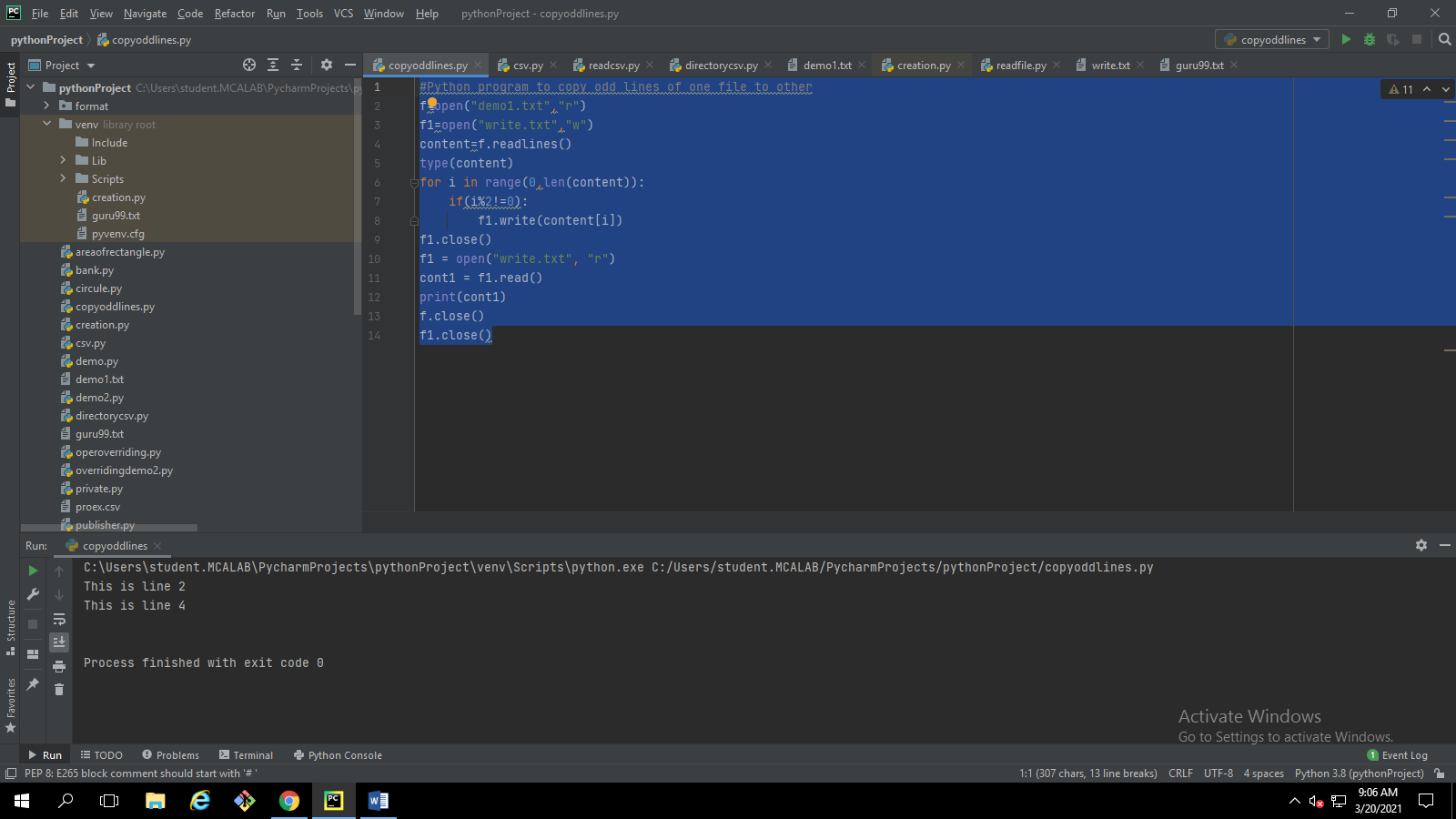
**CYCLE 6**

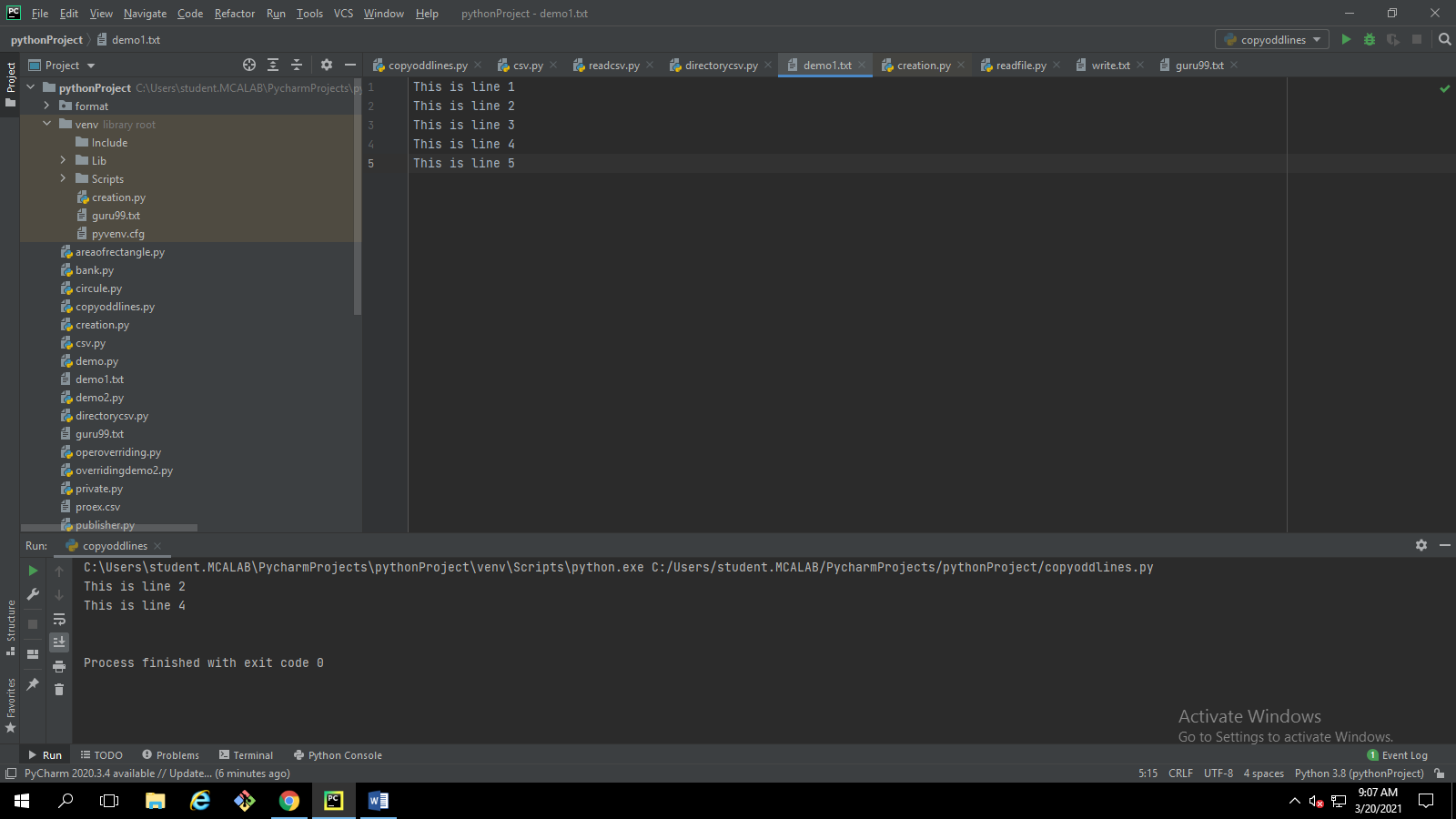
1. Write a Python program to read a file line by line and store it into a list
2. #Write a Python program to read a file line by line and store it into a list.  
   f=open("guru99.txt","r")  
   content=f.readlines()  
   print("file content is",content)  
   x=['anu baby']  
   x.append(content)  
   print("after appand line is",x)

**OUTPUT** 

Python program to copy odd lines of one file to other

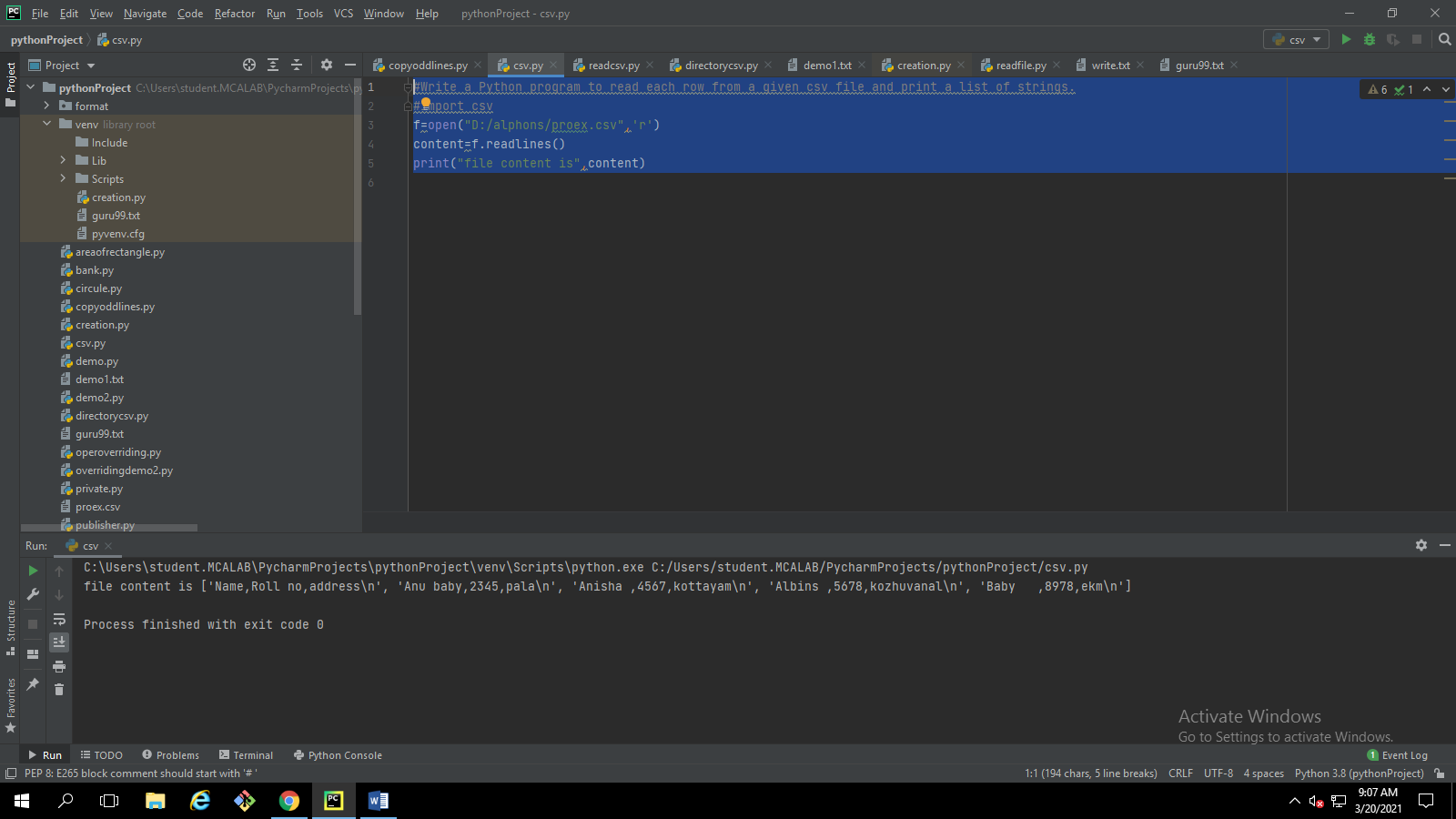
1. #Python program to copy odd lines of one file to other  
   f=open("demo1.txt","r")  
   f1=open("write.txt","w")  
   content=f.readlines()  
   type(content)  
   for i in range(0,len(content)):  
    if(i%2!=0):  
    f1.write(content[i])  
   f1.close()  
   f1 = open("write.txt", "r")  
   cont1 = f1.read()  
   print(cont1)  
   f.close()  
   f1.close()

**OUTPUT** 



Write a Python program to read each row from a given csv file and print a list of strings

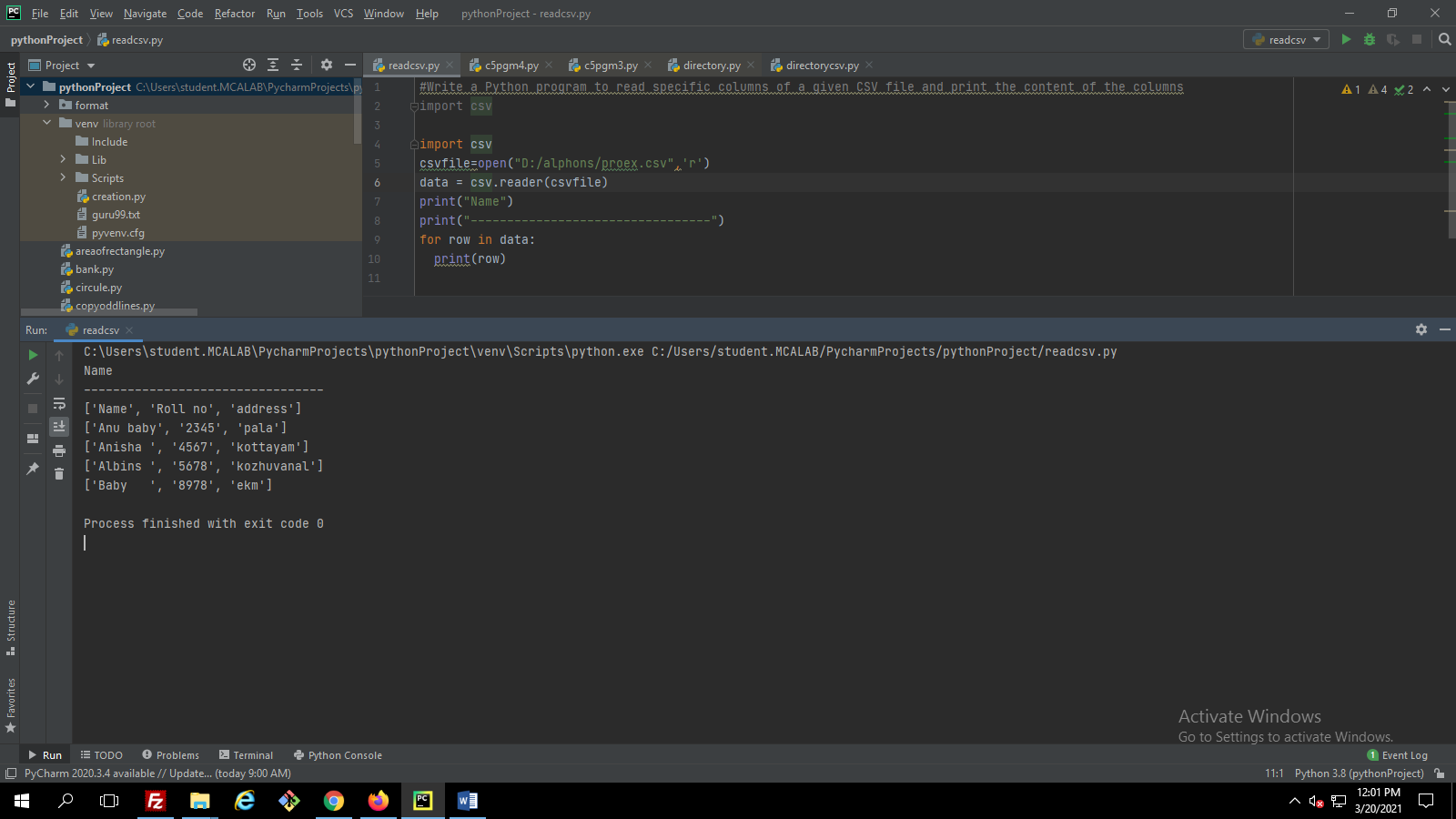
1. #Write a Python program to read each row from a given csv file and print a list of strings.  
   #import csv  
   f=open("D:/alphons/proex.csv",'r')  
   content=f.readlines()  
   print("file content is",content)

**OUTPUT** 

.Write a Python program to read specific columns of a given CSV file and print the content of the columns.

#Write a Python program to read specific columns of a given CSV file and print the content of the columns  
import csv  
  
import csv  
csvfile=open("D:/alphons/proex.csv",'r')  
data = csv.reader(csvfile)  
print("Name")  
print("---------------------------------")  
for row in data:  
 print(row)

**OUTPUT**



Write a Python program to write a Python dictionary to a csv file. After writing the CSV file read the CSV file and display the content

f = open("C:/Users/student.MCALAB/Desktop/details.csv", 'a')  
import json  
thisdict = {  
 "brand":"ford",  
 "model":"mustang",  
 "year":"1964"  
 }  
result =json.dumps(thisdict)  
f.write(result)  
f.close()  
f = open("C:/Users/student.MCALAB/Desktop/details.csv", 'r')  
print(f.read())

**OUTPUT** 