**COURSE3 OUTCOME 1**

**PROGRAM NO: 1**

**AIM** Familiarizing Text Editor, IDE, Code Analysis Tools etc // Use any IDE like PyCharm, PyDev…

An IDE (or Integrated Development Environment) is a program dedicated to software development. As the name implies, IDEs integrate several tools specifically designed for software development. These tools usually include:

* An editor designed to handle code (with, for example, syntax highlighting and auto-completion)
* Build, execution, and debugging tools
* Some form of source control

Most IDEs support many different programming languages and contain many more features. They can, therefore, be large and take time to download and install. You may also need advanced knowledge to use them properly.

**General Editors and IDEs with Python Support**

* **Eclipse + PyDev**

**Category:** IDE  
**Website:** [www.eclipse.org](http://www.eclipse.org/)  
**Python tools:** PyDev, [www.pydev.org](https://www.pydev.org/)

If you’ve spent any amount of time in the open-source community, you’ve heard about Eclipse. Available for Linux, Windows, and OS X at, Eclipse is the de-facto open-source IDE for Java development. It has a rich marketplace of extensions and add-ons, which makes Eclipse useful for a wide range of development activities.

One such extension is PyDev, which enables Python debugging, code completion, and an interactive Python console. Installing PyDev into Eclipse is easy: from Eclipse, select Help, Eclipse Marketplace, then search for PyDev. Click Install and restart Eclipse if necessary.

### Sublime Text

**Category:**CodeEditor  
 **Website:** [http://www.sublimetext.com](http://www.sublimetext.com/)

Written by a Google engineer with a dream for a better text editor, Sublime Text is an extremely popular code editor. Supported on all platforms, Sublime Text has built-in support for Python code editing and a rich set of extensions (called packages) that extend the syntax and editing features.Installing additional [Python packages](https://realpython.com/python-modules-packages/) can be tricky: all Sublime Text packages are written in Python itself, and installing community packages often requires you to execute Python scripts directly in Sublime Text.

### Visual Studio

**Category:** IDE  
**Website:** <https://www.visualstudio.com/vs/>  
**Python tools:** [Python Tools for Visual Studio](http://pytools.codeplex.com/), aka PTVS

Built by Microsoft, Visual Studio is a full-featured IDE, in many ways comparable to Eclipse. Built for Windows and Mac OS only, VS comes in both free (Community) and paid (Professional and Enterprise) versions. Visual Studio enables development for a variety of platforms and comes with its own marketplace for extensions.Python Tools for Visual Studio (aka PTVS) enables Python coding in Visual Studio, as well as Intellisense for Python, debugging, and other tools.

## Python-Specific Editors and IDEs

### PyCharm

**Category:** IDE  
**Website:** <https://www.jetbrains.com/pycharm/>

One of the best (and only) full-featured, dedicated IDEs for Python is [PyCharm](https://realpython.com/pycharm-guide/). Available in both paid (Professional) and free open-source (Community) editions, PyCharm installs quickly and easily on Windows, Mac OS X, and Linux platforms.Out of the box, PyCharm supports Python development directly. You can just open a new file and start writing code. You can run and debug Python directly inside PyCharm, and it has support for source control and projects.

**PROGRAM NO: 2**

**AIM:** Display future leap years from current year to a final year entered by user.

**ALGORITHM:**

Step1: Take current year and final year as inputs

Step2: if current year < final year

Step3: Check for leap year condition

Step4: Then print list of leap years between current and final year

**PROGRAM:**

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

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

if current < final:

print ("Here is a list of leap years between " + str(current) + " and " + str(final) + ":")

while current < final:

if current % 4 == 0:

print(current)

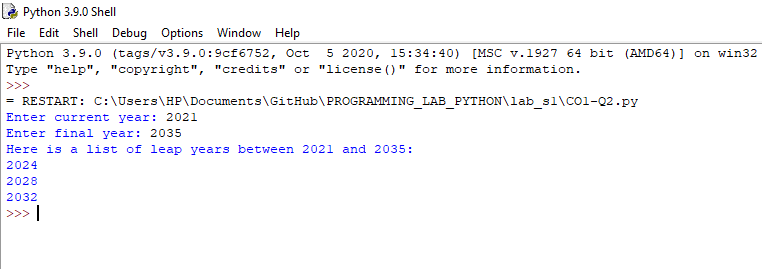
if current % 100 == 0 and current % 400 == 0:

print(current)

current += 1

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT:**

****

**PROGRAM NO: 3**

**AIM**: 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)

**ALGORITHM:**

1. Step1: Take in the number of elements to be in the list from the user.

Step2: Using a for in list comprehensions, get the elements one by one from the list and check if it is positive

Step3: If it is positive, print the numbers as a list

1. Step1:

**PROGRAM:**

a)

list1 = [1,-1, -21, 0, 45, 66,2,-3,4,-6, -93]

print(list1)

a=[num for num in list1 if num>=0]

print("Positive integers in the list are:",a)

b)

N=int(input("Enter limit N:"))

x = (x\*\*2 for x in range(N))

x = list(x)

print(x)

c)

V =[' a', 'e', 'i', 'o', 'u', 'A', 'E', 'I', 'O', 'U' ]

print("V=['a', 'e', 'i', 'o', 'u'', ‘A', 'E', 'I', 'O', 'U' ]")

w=str(input("Enter the word: "))

x = [x for x in w if any([v in x for v in V])]

x = list(x)

print ("Vowels in given word:",x)

d) a=str(input("Enter word:"))

a=list(a)

x=[ord(x) for x in a ]

x=list(x)

print(x)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

**PROGRAM NO: 4**

**AIM**: Count the occurrences of each word in a line of text.

**ALGORITHM:**

**PROGRAM:**

a=str(input("Enter word:"))

print(a)

s=a.split(' ')

count = {}

print

for n in s:

count[n]=count.get(n,0)+1

print("The occurrences of each word in a given line is :")

print(count)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

**PROGRAM NO: 5**

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

**ALGORITHM**

**PROGRAM:**

lst = [ ]

lst = [int(item) for item in input("Enter the list items : ").split()]

print("INPUT IS",lst)

x= ["over" if x>100 else x for x in lst]

lst=list(x)

print(lst)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

**PROGRAM NO: 6**

**AIM:** Store a list of first names. Count the occurrences of ‘a’ within the list

**ALGORITHM**

PROGRAM:

lst=['anu','ammu','ananya']

print(lst)

i=0

count=0

while i<len(lst):

count=count+lst[i].count('a')

i=i+1

print("Count of a is: " ,count)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

**PROGRAM NO: 7**

**AIM:** 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

**ALGORITHM**

**PROGRAM:**

lst1=[4,5,3,2,1]

lst2=[8,4,3,2,1,5,9]

print("lst1=",lst1)

print("lst2=",lst2)

a=len(lst1)

b=len(lst2)

if a==b:

print("SAME LENGTH")

else:

print("NOT SAME LENGTH")

s1=sum(lst1)

s2=sum(lst2)

if s1==s2:

print("SUM IS SAME")

else:

print("SUM IS NOT SAME")

lst1=set(lst1)

lst2=set(lst2)

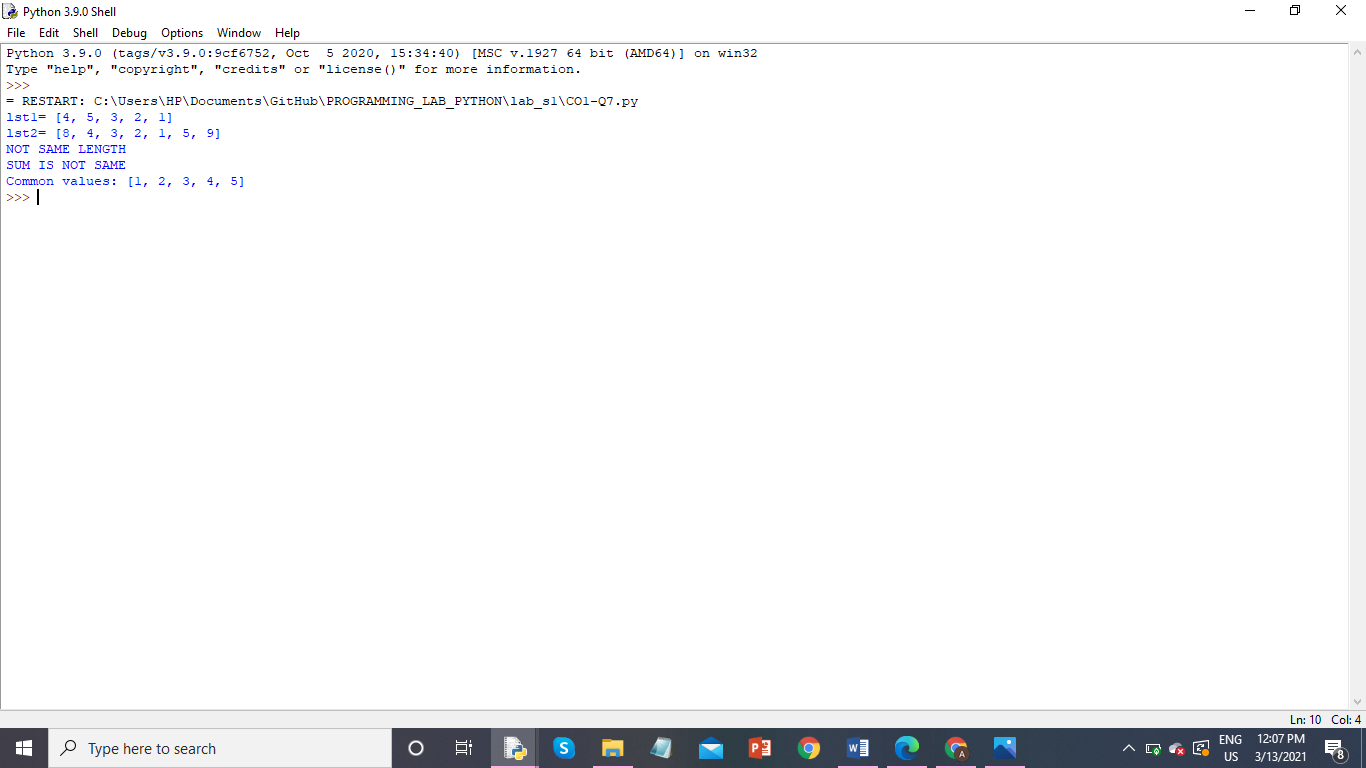
i = lst1.intersection(lst2)

i=list(i)

print("Common values:",i)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**PROGRAM NO: 8**

**AIM:** Get a string from an input string where all occurrences of first character replaced with ‘$’, except first character.

**ALGORITHM**

**PROGRAM:**

str1=input('Enter string ')

print('input sring is ',str1)

char = str1[0]

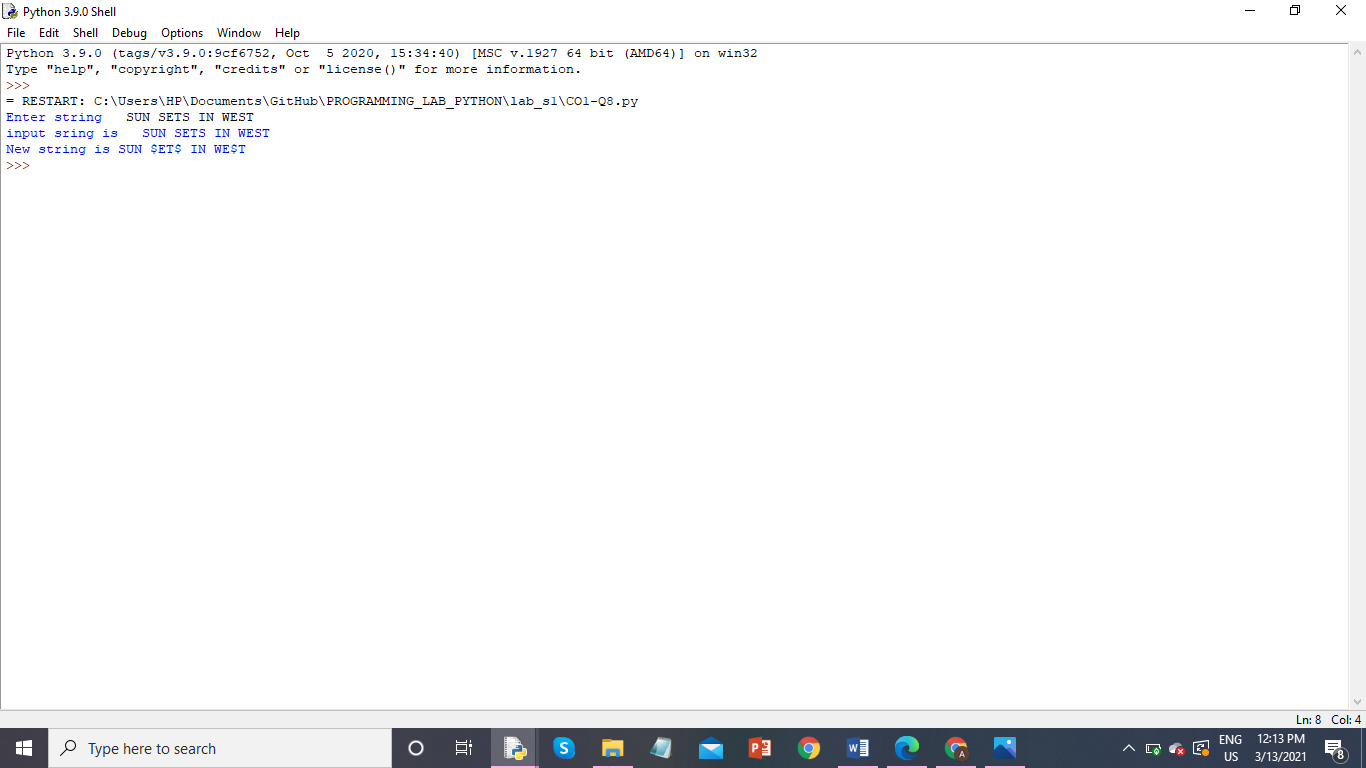
str1 = str1.replace(char, '$')

str1 = char + str1[1:]

print('New string is',str1)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**PROGRAM NO: 9**

**AIM:** Create a string from given string where first and last characters exchanged.

**ALGORITHM**

**PROGRAM:**

s = str(input("Enter the String:"))

print("INPUT IS:",s)

slice\_mid=s[1:-1]

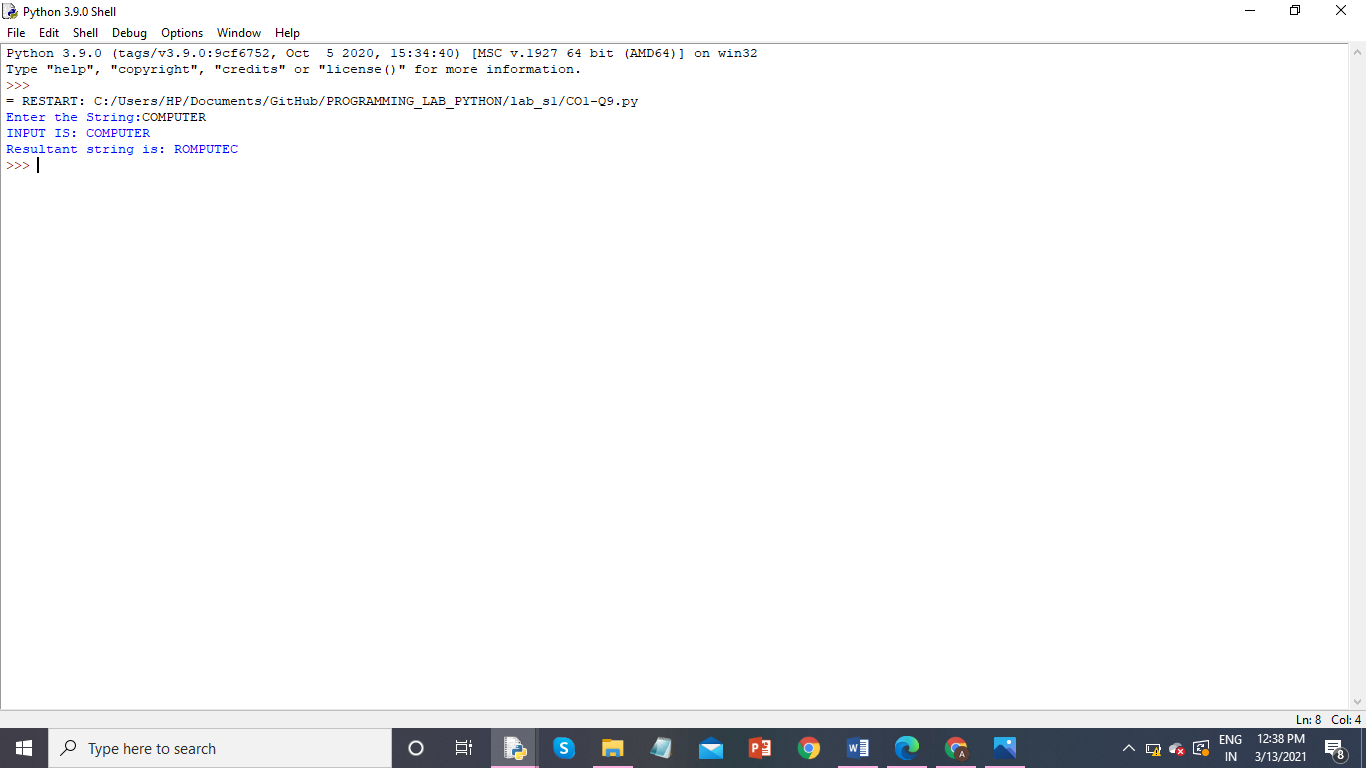
slice\_beg=s[0]

slice\_end=s[-1:]

print("Resultant string is:",slice\_end+slice\_mid+slice\_beg)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**PROGRAM NO: 10**

**AIM:** Accept the radius from user and find area of circle.

**ALGORITHM**

**PROGRAM:**

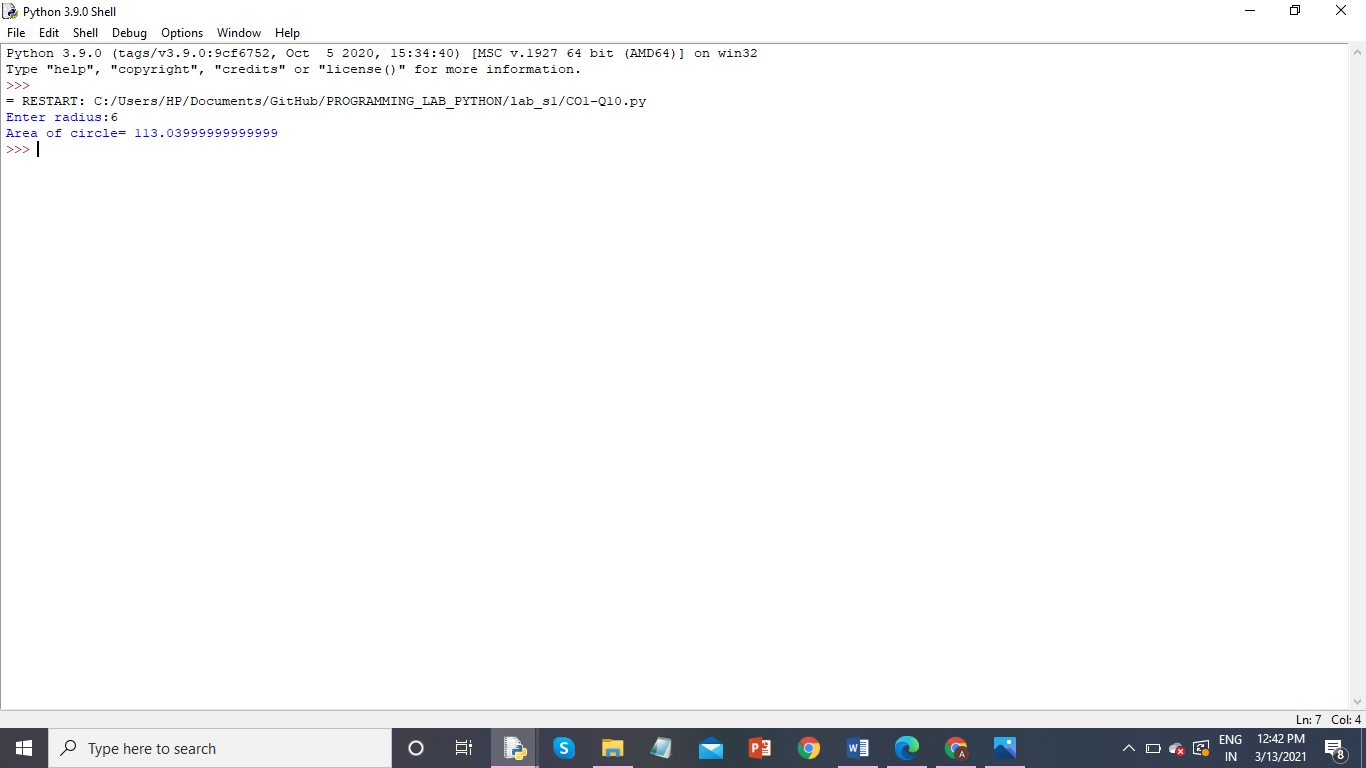
pi=3.14

R=int(input("Enter radius:"))

print("Area of circle=",pi\*R\*R)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**PROGRAM NO: 11**

**AIM:** Find biggest of 3 numbers entered.

**ALGORITHM**

**PROGRAM:**

n1 = float(input("Enter num 1:"))

n2 = float(input("Enter num 2:"))

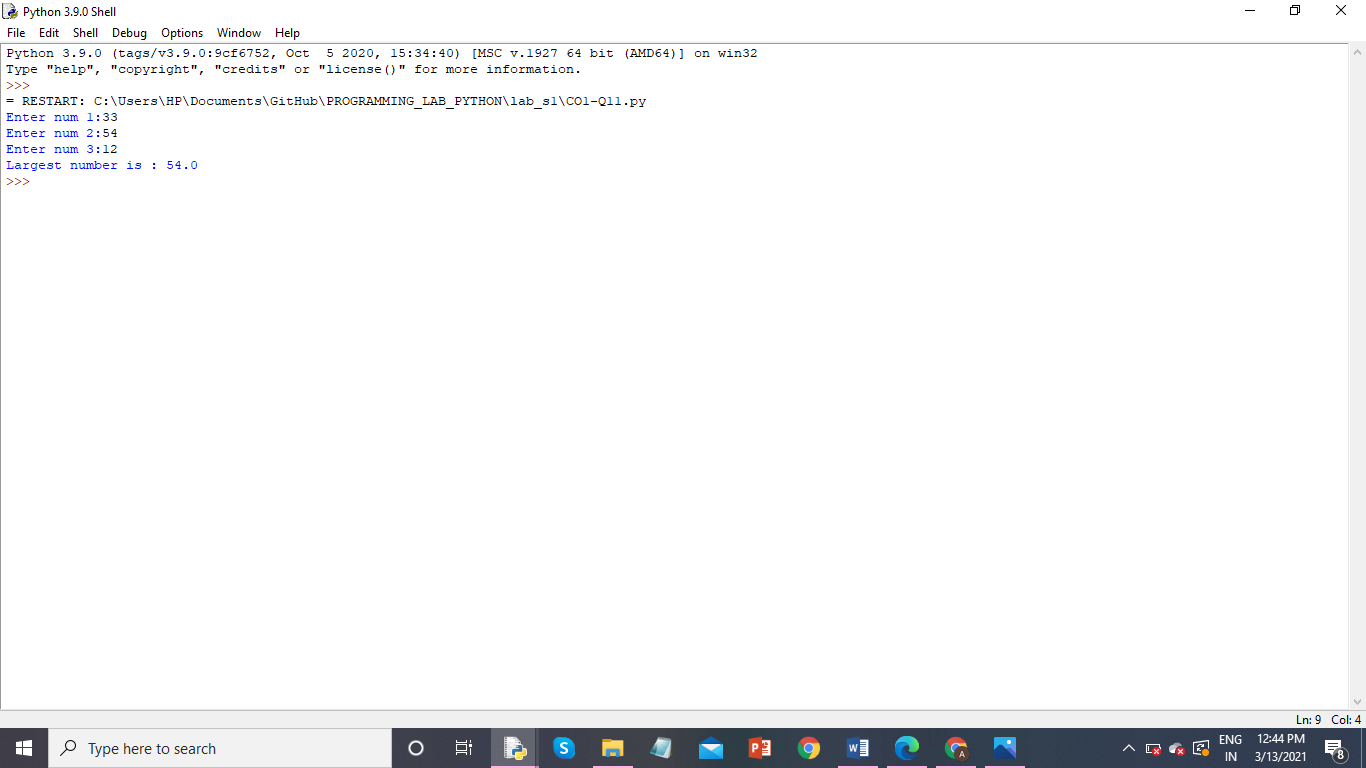
n3 = float(input("Enter num 3:"))

large=max(n1 , n2 , n3)

print("Largest number is :",large)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**PROGRAM NO: 12**

**AIM:** Accept a file name from user and print extension of that

**ALGORITHM**

**PROGRAM:**

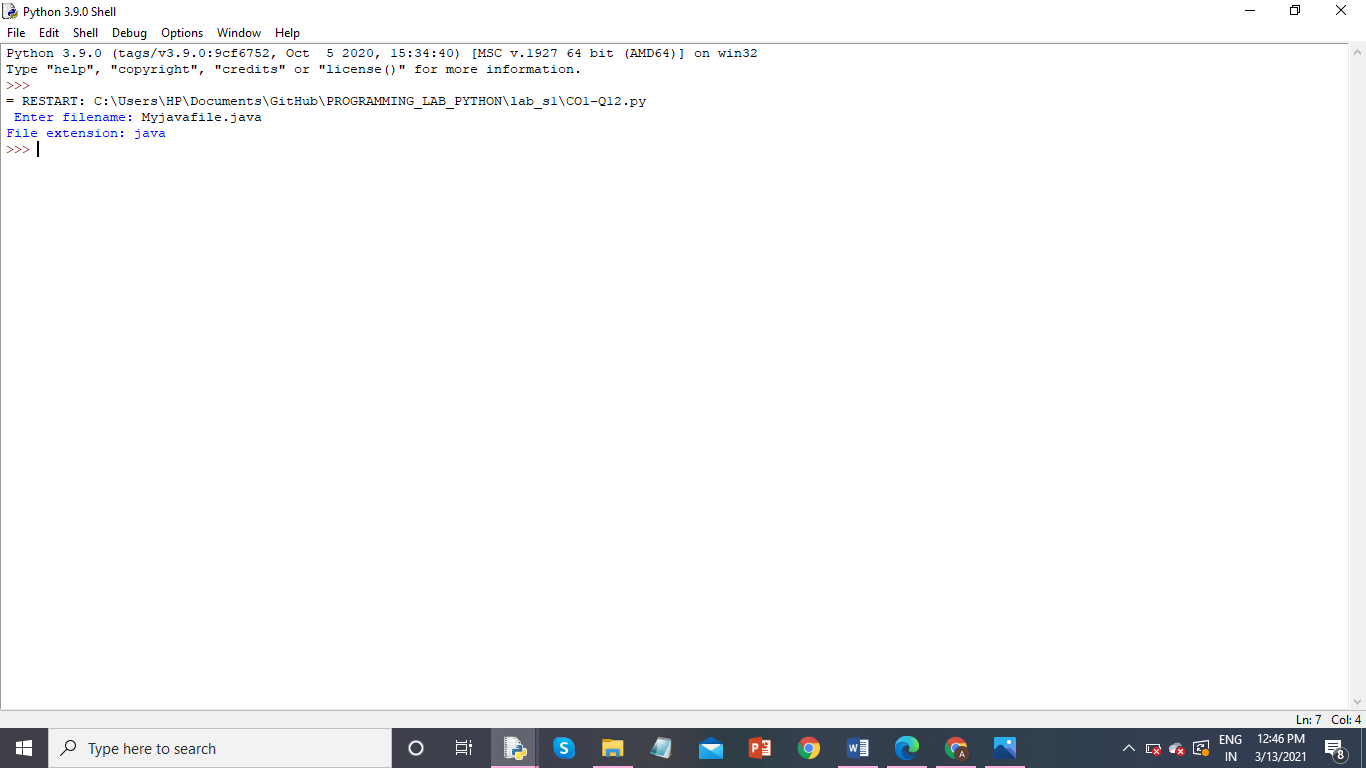
str=input(" Enter filename: ")

t=str.split('.')

print("File extension: " +t[-1])

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**PROGRAM NO: 13**

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

**ALGORITHM**

**PROGRAM:**

color\_lst=["red","blue","black","white","yellow","orange"]

print(color\_lst)

a = color\_lst[0]

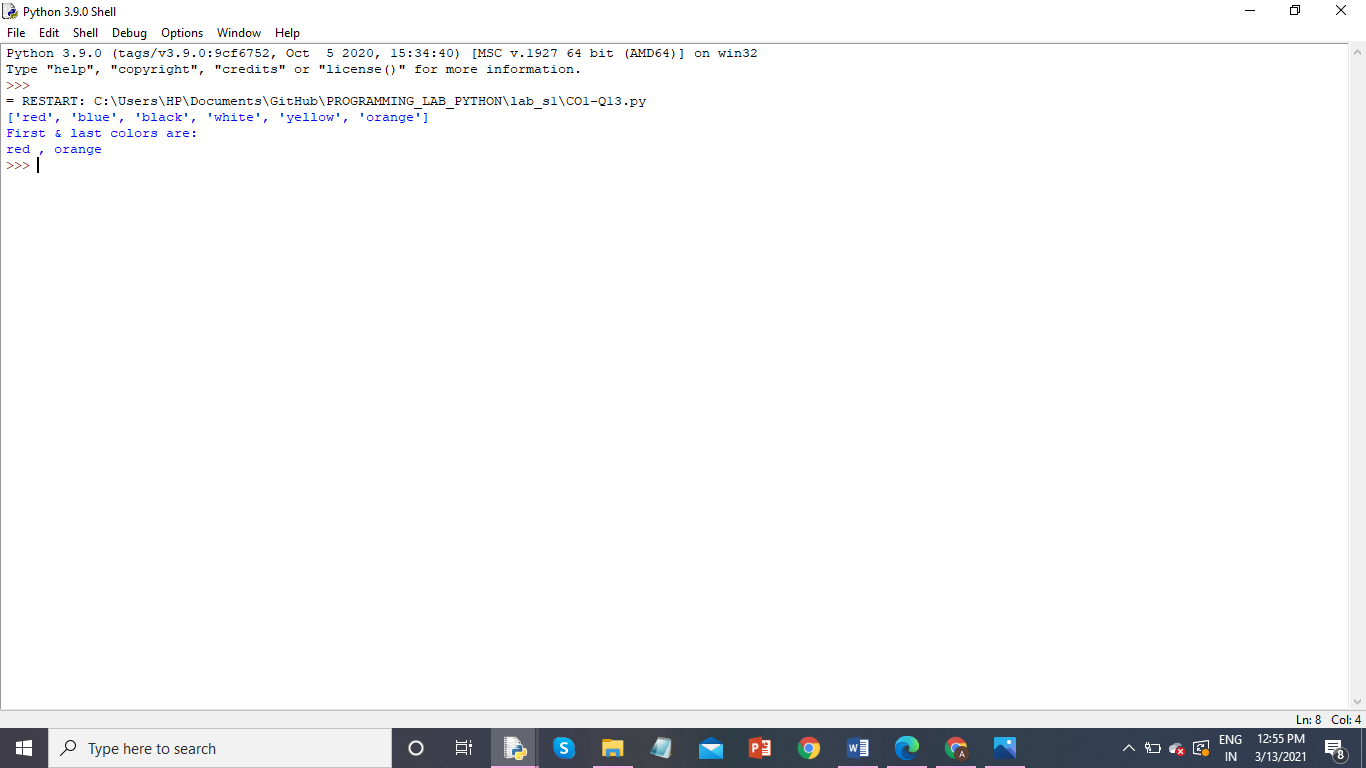
b= color\_lst[-1]

print("First & last colors are:")

print(a,b, sep = " , ")

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**PROGRAM NO: 14**

**AIM:** Accept an integer n and compute n+nn+nnn.

**ALGORITHM**

**PROGRAM:**

N = int(input("Enter the integer N :"))

tmp = N

tmp1 = tmp\*tmp

tmp2 = tmp\*tmp\*tmp

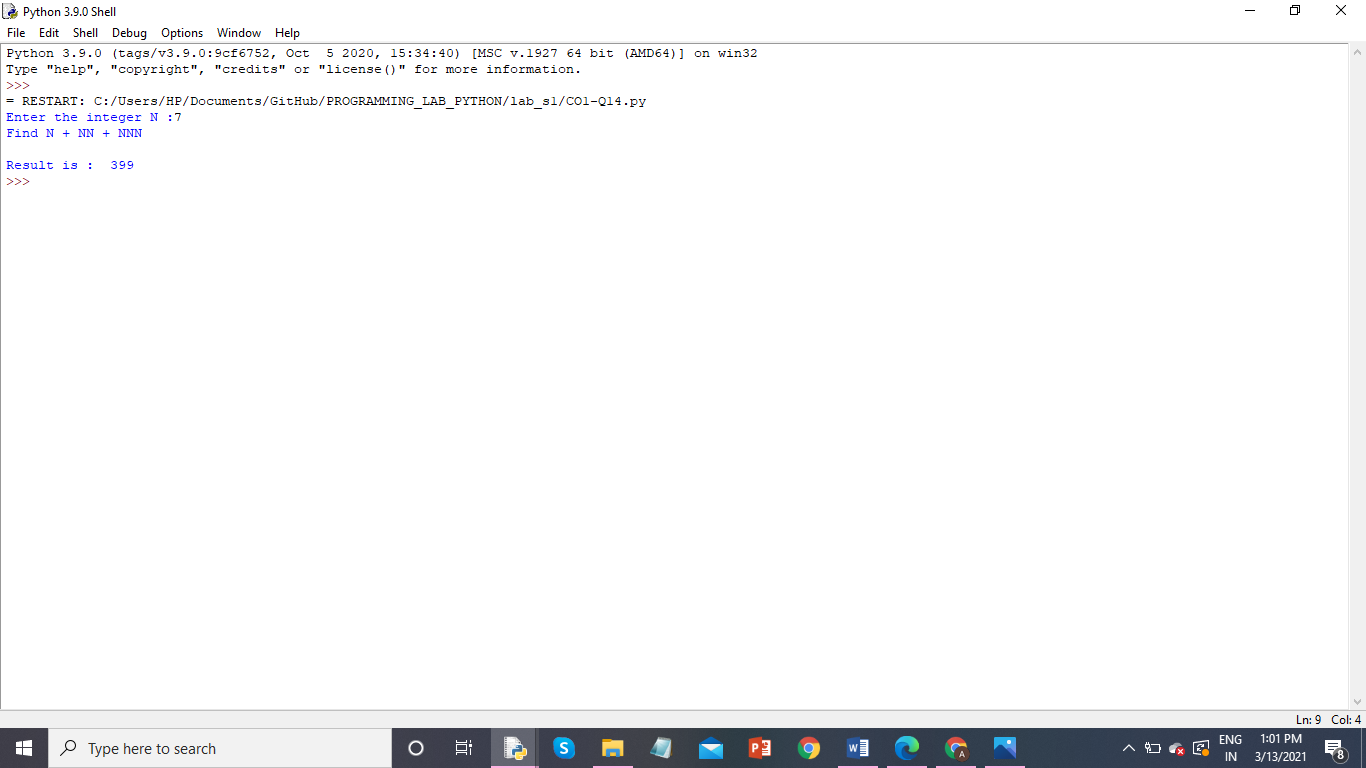
print("Find N + NN + NNN\n")

comp = tmp + tmp1 + tmp2

print("Result is : ",comp)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**PROGRAM NO: 15**

**AIM:** Print out all colors from color-list1 not contained in color-list2

**ALGORITHM**

**PROGRAM:**

colorlist1=set(['orange','green','blue','violet','pink','white'])

print(colorlist1)

colorlist2=set(['white','blue','violet'])

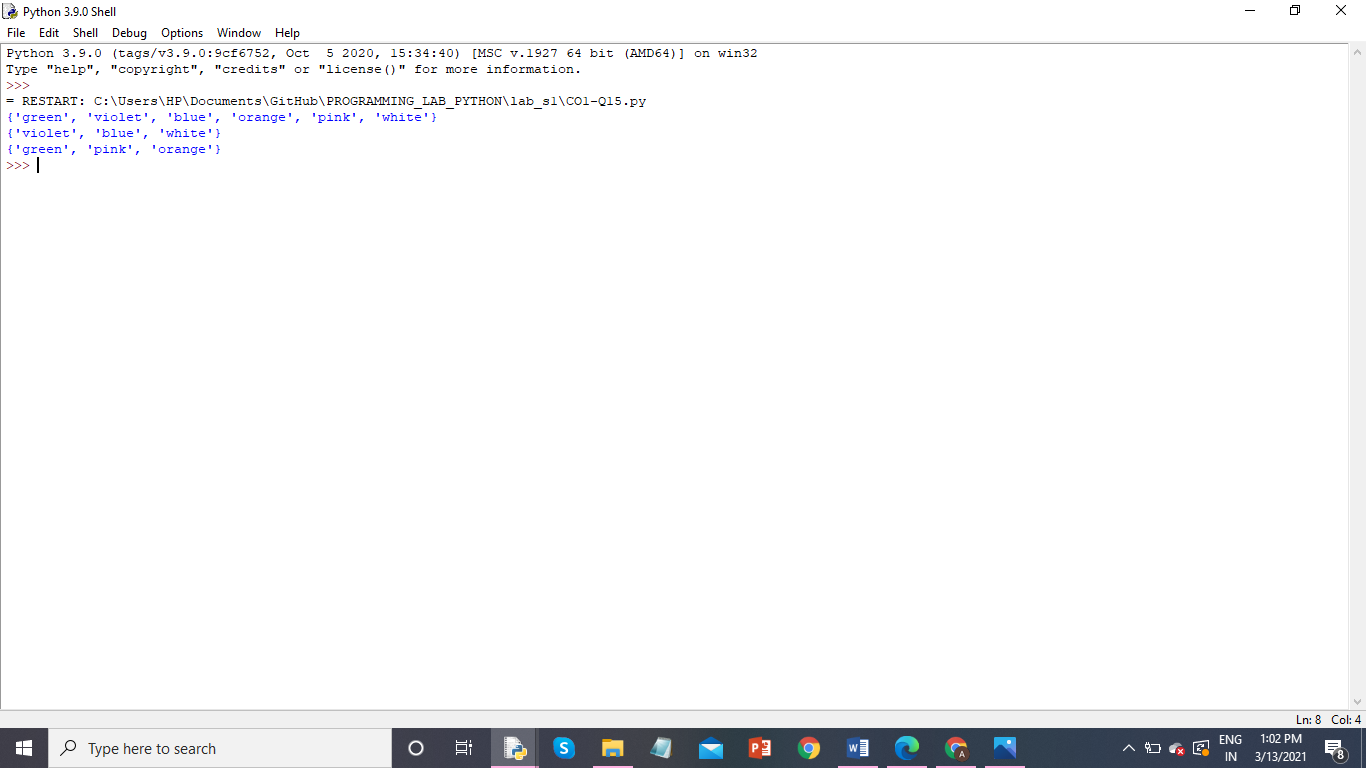
print(colorlist2)

a=(colorlist1.difference(colorlist2))

print(a)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**PROGRAM NO: 16**

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

**ALGORITHM**

**PROGRAM:**

a= "PYTHON"

b="JAVA"

print ("a=",a)

print("b=",b)

print(a +" " + b )

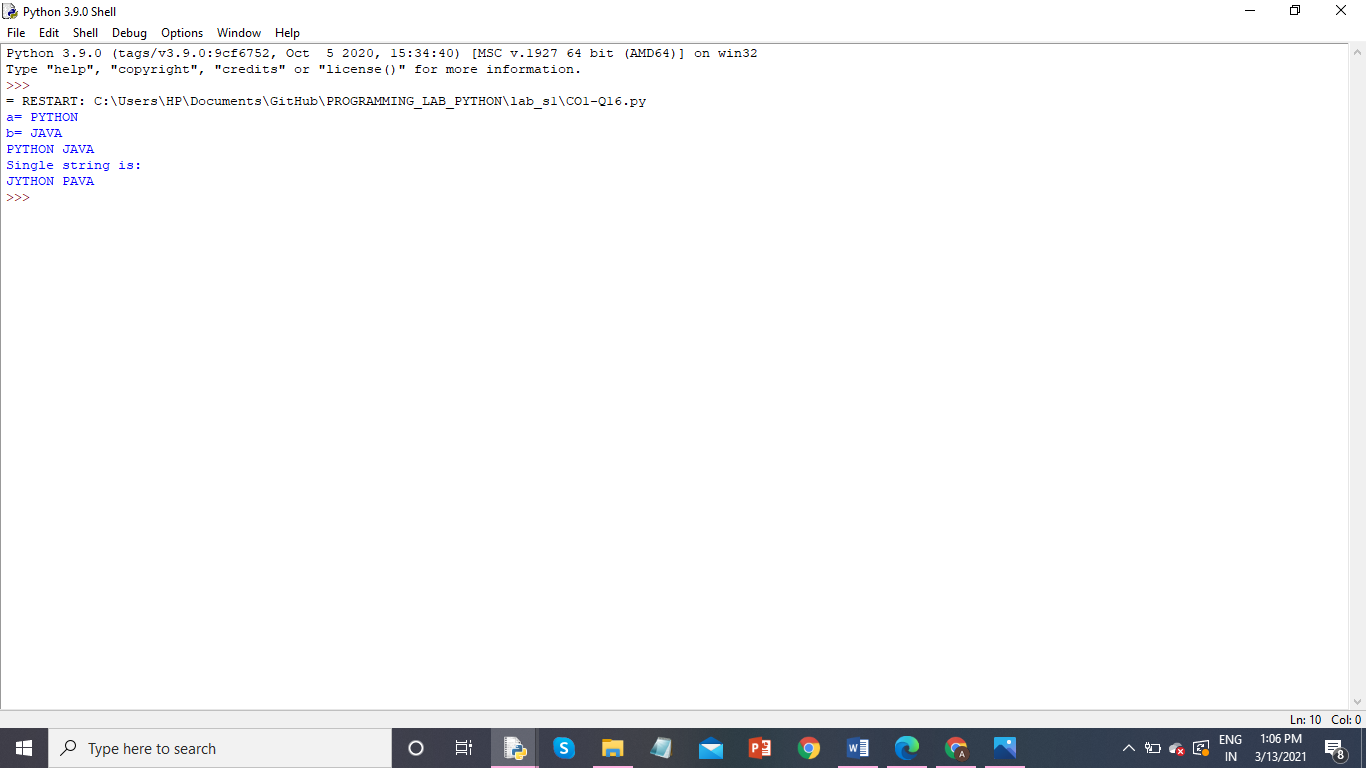
a1 = b[:1] + a[1:]

b2= a[:1] + b[1:]

print("Single string is:")

print(a1 + " "+ b2)

**RESULT:** The above program is successfully executed and obtained the output



**OUTPUT**

**PROGRAM NO: 17**

**AIM:** Sort dictionary in ascending and descending order.

**ALGORITHM**

**PROGRAM:**

a1 ={'Swathi':67,'Anu':98,'Riya':66,'Vismaya':88,'Neema':75,'Reshma':89}

print("Inputed dict is :", a1)

a1\_sorted\_keys = sorted(a1, key=a1.get, reverse=True)

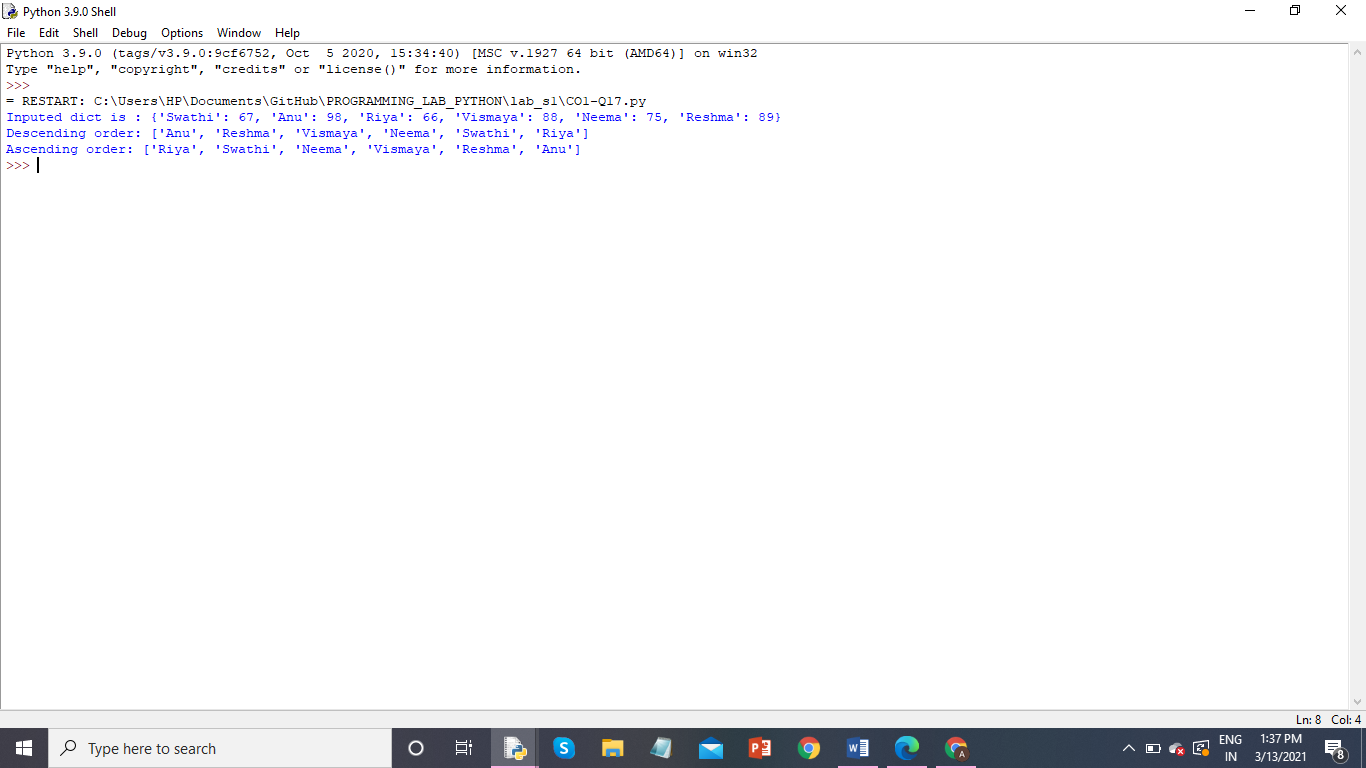
a1\_sorted\_keys\_2 = sorted(a1, key=a1.get)

print("Descending order:",a1\_sorted\_keys)

print("Ascending order:",a1\_sorted\_keys\_2)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**PROGRAM NO: 18**

**AIM:** Merge two dictionaries

**ALGORITHM**

**PROGRAM:**

def Merge(dict1, dict2):

return (dict2.update(dict1))

dict1 = {'a': 100, 'b': 48, 'e': 55}

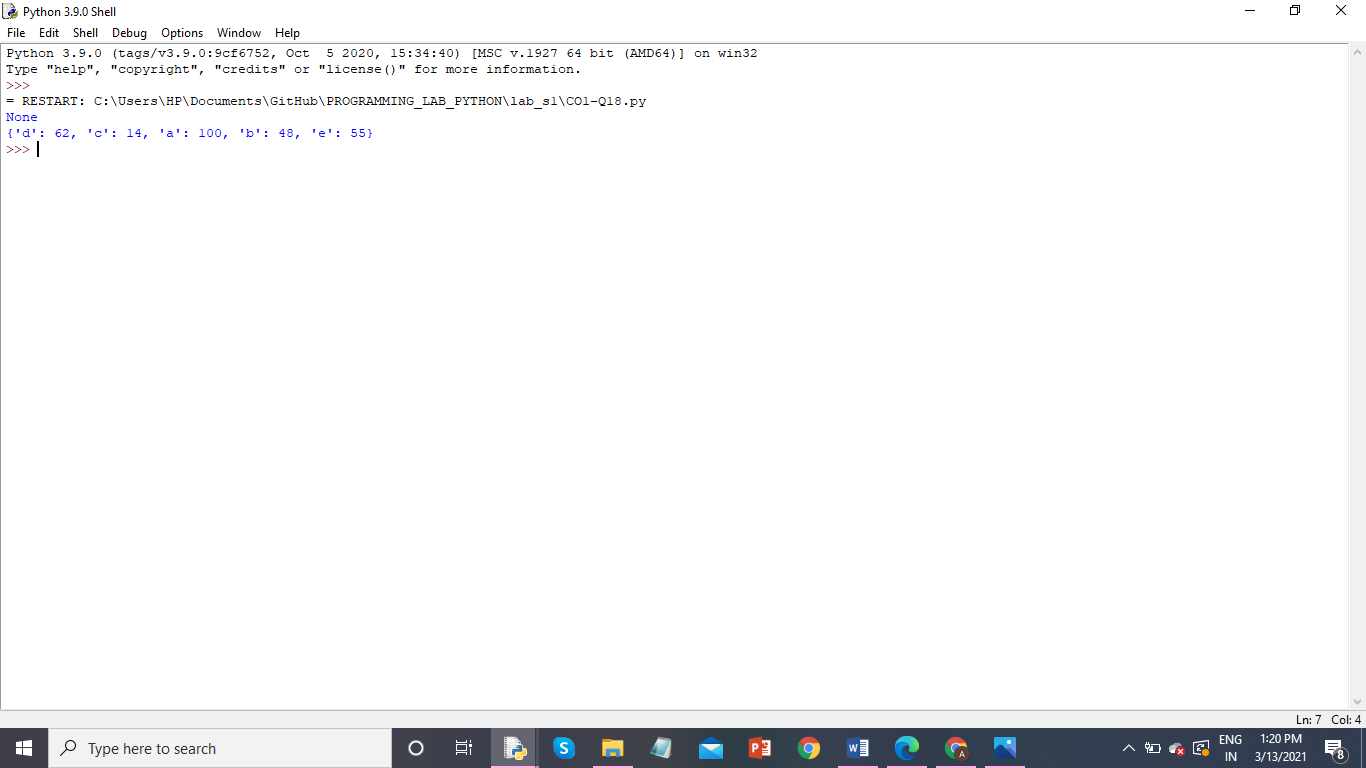
dict2 = {'d': 62, 'c': 14}

print(Merge(dict1, dict2))

print(dict2)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**PROGRAM NO: 19**

**AIM:** Find gcd of 2 numbers.

**ALGORITHM**

**PROGRAM:**

n1=int(input("ENTER FIRST NUM:"))

n2=int(input("ENTER SECOND NUM:"))

gcd=1

if n1%n2==0:

print(n2)

for k in range(int(n2 / 2), 0, -1):

if n1 % k == 0 and n2 % k == 0:

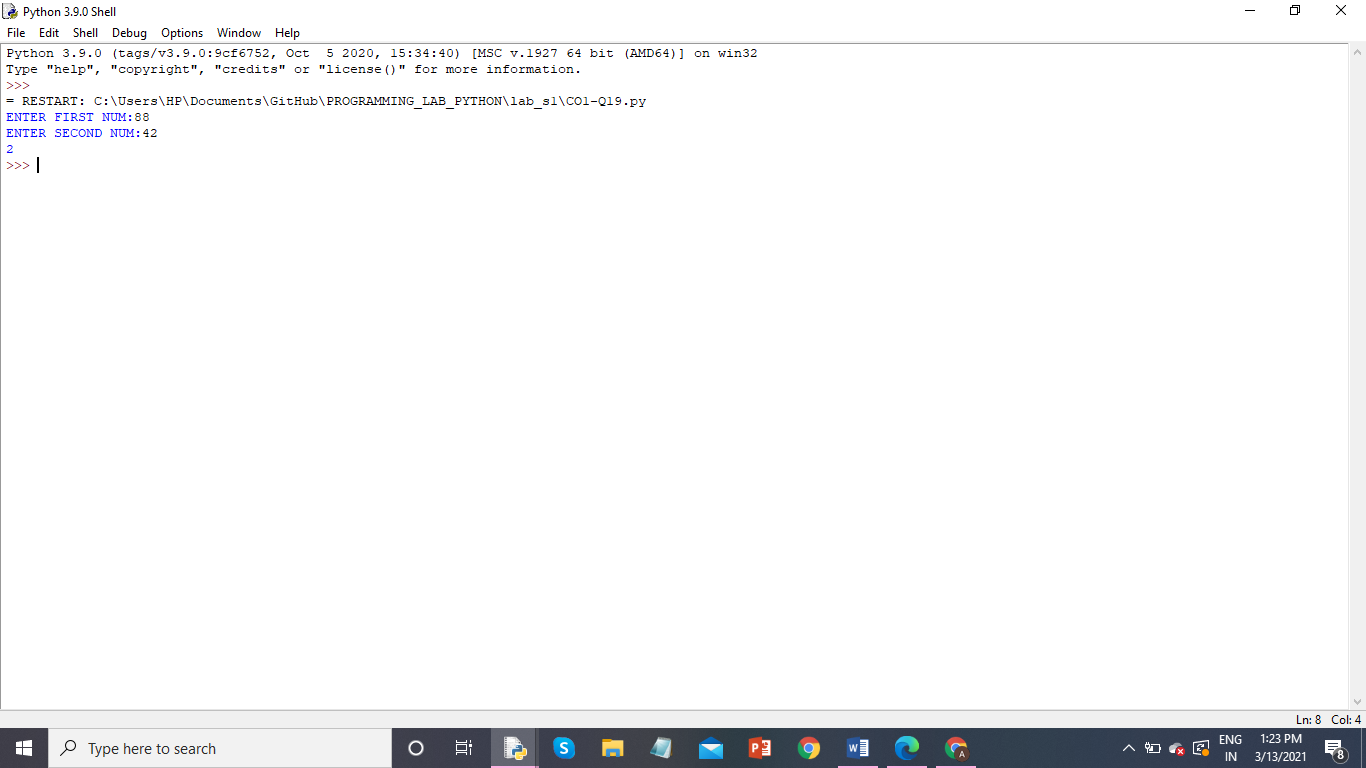
gcd = k

break

print(k)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**PROGRAM NO: 20**

**AIM:** From a list of integers, create a list removing even numbers.

**ALGORITHM**

**PROGRAM:**

li=[33,88,9,12,45,78,11,77]

print("ORIGINAL LIST:",li)

for i in li:

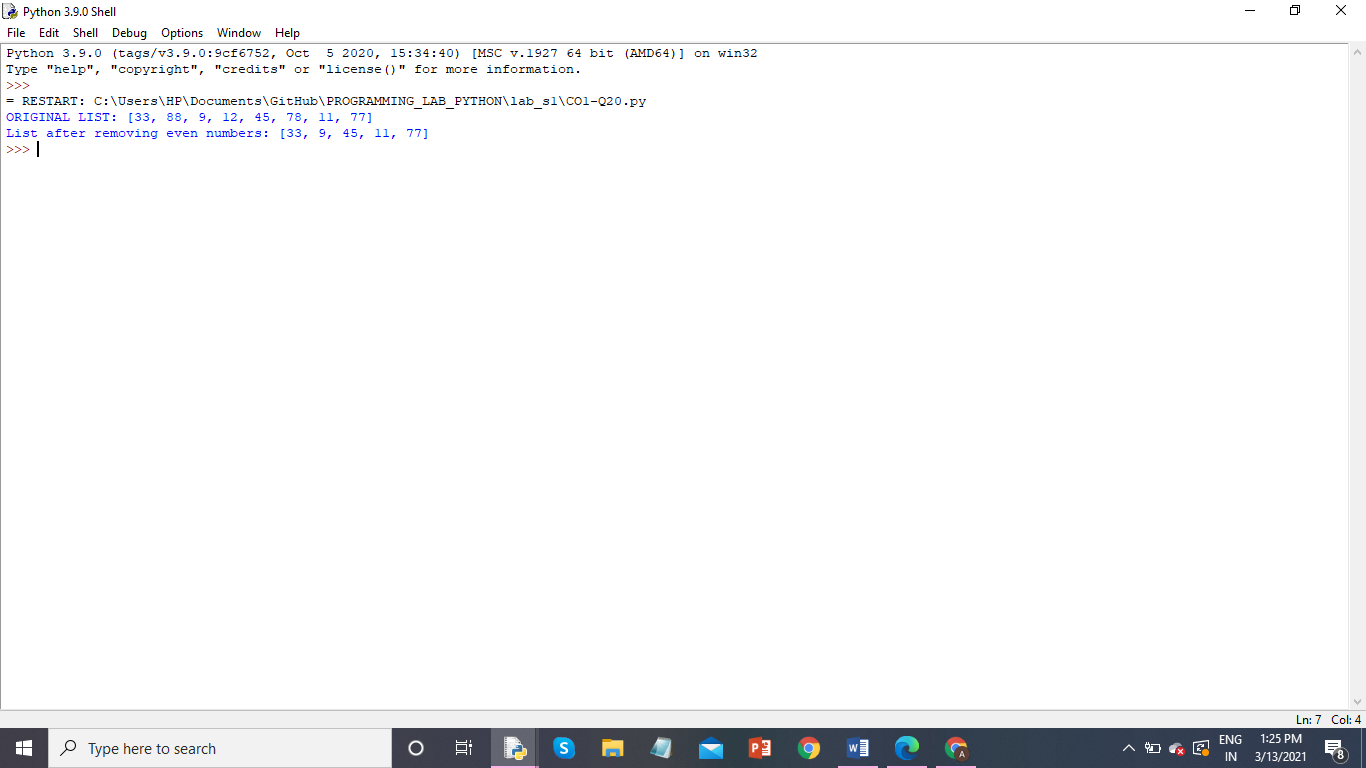
if (i % 2 ==0):

li.remove(i)

print("List after removing even numbers:",li)

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**

****

**COURSE OUTCOME 2**

**PROGRAM NO: 1**

**AIM:**

**ALGORITHM**

**PROGRAM**

**RESULT:** The above program is successfully executed and obtained the output

**OUTPUT**