

PRACTICAL JOURNAL OF **PROGRAMMING PRACTICES**

**BTech: Second-Year**

|  |  |  |
| --- | --- | --- |
| **Name of Student** | : | **KIRTI RATHORE** |
| **Branch & Section** | **:** | **CSE & CS-02.** |
| **Roll Number** | **:** | **0827CS193D09** |
| **Year** | **:** | **2nd** |

**Department of COMPUTER SCIENCE & ENGINEERING**

**AITR, Indore,**

**ACROPOLIS INSTITUTE OF TECHNOLOGY & RESEARCH, INDORE**

**Department of COMPUTER SCIENCE & ENGINEERING**

**Certificate**

This is to certify that the experimental work entered in this journal as per the BE II year syllabus prescribed by the RGPV was done by Mr. **KIRTI RATHORE** Btech 2ndYear4thSemester in the OOPM Laboratory of this institute during the academic year 2019– 2020.

Signature of Head Signature of the Faculty

**GENERAL INSTRUCTIONS FOR LABORATORY CLASSES**

**DO’S**

* Without Prior permission do not enter into the Laboratory.
* While entering into the LAB students should wear their ID cards.
* The Students should come with proper uniform.
* Students should sign in the LOGIN REGISTER before entering into the laboratory.
* Students should come with observation and record note book to the laboratory.
* Students should maintain silence inside the laboratory.
* After completing the laboratory exercise, make sure to shutdown the system properly.

**DONT’S**

* Students bringing the bags inside the laboratory.
* Students using the computers in an improper way.
* Students scribbling on the desk and mishandling the chairs.
* Students using mobile phones inside the laboratory.
* Students making noise inside the laboratory.

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Computer Science and Engineering, IV-Semester

**CS406 Programming Practices (c) Python**

Introduction: Basic syntax, Literal Constants, Numbers, Variable and Basic data types,

String, Escape Sequences, Operators and Expressions, Evaluation Order, Indentation, Input

Output, Functions, Comments.

Data Structure: List, Tuples, Dictionary and Sets.

Control Flow: Conditional Statements - If, If-else, Nested If-else. Iterative Statement -

For, While, Nested Loops. Control statements - Break, Continue, Pass.

Object oriented programming: Class and Object, Attributes, Methods, Scopes and

Namespaces, Inheritance, Overloading, Overriding, Data hiding.

Exception: Exception Handling, Except clause, Try finally clause, User Defined

Exceptions.

Modules and Packages

Standard Libraries: File I/0, Sys, logging, Regular expression, Date and Time, Network

programming, multi-processing and multi-threading.

References

 Timothy A. Budd: Exploring python, McGraw-Hill Education.

 R.Nageshwar Rao ,”Python Programming” ,Wiley India

 Think Python: Allen B. Downey, O'Reilly Media, Inc.

**INDEX**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No.** | **Name of Experiment** | **Page No.** | **Faculty**  **Sign** |
| A. | Introduction to Python |  |  |
| B. | How to download python? |  |  |
| C. | How to install python? |  |  |
| 1. | Armstrong Number |  |  |
| 2. | Factorial |  |  |
| 3. | Fibonacci series |  |  |
| 4. | Check if number is even or odd |  |  |
| 5. | Calculate Gross salary |  |  |
| 6. | Check if leap year |  |  |
| 7. | Reverse of a number |  |  |
| 8. | Sum of digits of a number |  |  |
| 9. | Swapping of 2 numbers using third variable |  |  |
| 10. | Swapping of 2 numbers without using third variable |  |  |
| 11. | Check if character is uppercase or lowercase |  |  |
| 12. | Count number of characters in string |  |  |
| 13. | String with maximum length in a list |  |  |
| 14. | Occurrence of all characters in a string |  |  |
| 15. | Happy number |  |  |
| 16. | Print the string in sorted order |  |  |
| 17. | Declaration of Function (1) |  |  |
| 18. | Declaration of Function (2) |  |  |
| 19. | Declaration of Function (3) |  |  |
| 20. | Declaration of Function (4) |  |  |
|  |  |  |  |
|  | **BASIC PROGRAMS** |  |  |
| 1 | Write a python program to print "Hello". |  |  |
| 2 | Write a python program to add, subtract, multiply & divide two numbers. |  |  |
| 3 | Write a python program to find percentage of students. |  |  |
| 4 | Write a python program to swap two numbers using a temporary variables. |  |  |
| 5 | Write a python program to swap two numbers without using a temporary variable |  |  |
| 6 | Write a python program to calculate factorial of a number. |  |  |
| 7 | Write a python program to take user input. |  |  |
|  | **IF-ELSE** |  |  |
| 8 | Write a python program to accept two integers and check whether they are equal or not |  |  |
| 9 | Write a python program to program to find even and odd number. |  |  |
| 10 | Write a python program to find greatest among three numbers. |  |  |
| 11 | Write a python program to check whether a given number is positive or negative |  |  |
| 12 | Write a python program to check whether entered character is vowel or consonant. |  |  |
| 13 | Write a python program to check result of student(Pass or Fail) |  |  |
| 14 | Write a python program to find whether a given year is a leap year or not |  |  |
|  |  |  |  |
|  | **LOOPS** |  |  |
| 15 | Write a python program to display numbers from 1 to 100 using a loop |  |  |
| 16 | Write a python program to display table of a number taken from user input. |  |  |
| 17 | Write a python program to print factorial of a number received as a user input. |  |  |
| 18 | Write a python program to find Fibonacci series up to n terms |  |  |
| 19 | Write a python program to generate prime numbers from 1 to 50. |  |  |
|  |  |  |  |
|  | **LIST** |  |  |
| 20 | Write a python program to create a list. |  |  |
| 21 | Write a python program to access elements of a list. |  |  |
| 22 | Write a python program to print even numbers present in a list. |  |  |
| 23 | Write a python program to multiplies all the items in a list. |  |  |
| 24 | Write a python program to get the smallest number from a list. |  |  |
| 25 | Write a python program to get the largest number from a list. |  |  |
|  |  |  |  |
|  | **TUPLE** |  |  |
| 26 | Write a python program to create a tuple. |  |  |
| 27 | Write a python program to create a tuple with numbers and print one item |  |  |
| 28 | Write a python program to add an item in a tuple |  |  |
| 29 | Write a python program to convert a list to a tuple |  |  |
| 30 | Write a python program to find the length of a tuple |  |  |
|  |  |  |  |
|  | **SET** |  |  |
| 31 | Write a Python program to create a set |  |  |
| 32 | Write a Python program to add member(s) in a set |  |  |
| 33 | Write a Python program to remove item(s) from set |  |  |
| 34 | Write a Python program to create an intersection of sets |  |  |
| 35 | Write a Python program to create a union of sets |  |  |
| 36 | Write a Python program to create set difference |  |  |
| 37 | Write a Python program to clear a set |  |  |
|  |  |  |  |
|  | **DICTIONARY** |  |  |
| 38 | Write a Python program to create a dictionary |  |  |
| 39 | Write a Python program t to add a key to a dictionary |  |  |
| 40 | Write a Python program to check whether a given key already exists in a dictionary. |  |  |
| 41 | Write a Python program to merge two Python dictionaries |  |  |
| 42 | Write a Python program to sum all the items in a dictionary |  |  |
|  |  |  |  |
|  | **FUNCTION** |  |  |
| 43 | Write a Python function to find average of five numbers passed parameters. |  |  |
| 44 | Write a Python function to take list as argument and remove odd numbers from list. |  |  |
| 45 | Write a Python function to find Fibonacci series up to n terms |  |  |
| 46 | Write a Python function to find factorial of a number using function |  |  |
| 47 | Write a Python function to generate a table of any number entered by user. |  |  |
| 48 | Write a Python function to check whether given number is prime or composite(using function) |  |  |
| 49 | Write a Python function to generate prime numbers from 1 to 100 (using function) |  |  |
| 50 | Write a Python function to check whether input number is even or odd. |  |  |
| 51 | Write a Python function to sum all the numbers in a list |  |  |

Introduction to Python

What is python?

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

* web development (server-side),
* software development,
* mathematics,
* system scripting.

What can Python do?

* Python can be used on a server to create web applications.
* Python can be used alongside software to create workflows.
* Python can connect to database systems. It can also read and modify files.
* Python can be used to handle big data and perform complex mathematics.
* Python can be used for rapid prototyping, or for production-ready software development.

Why Python?

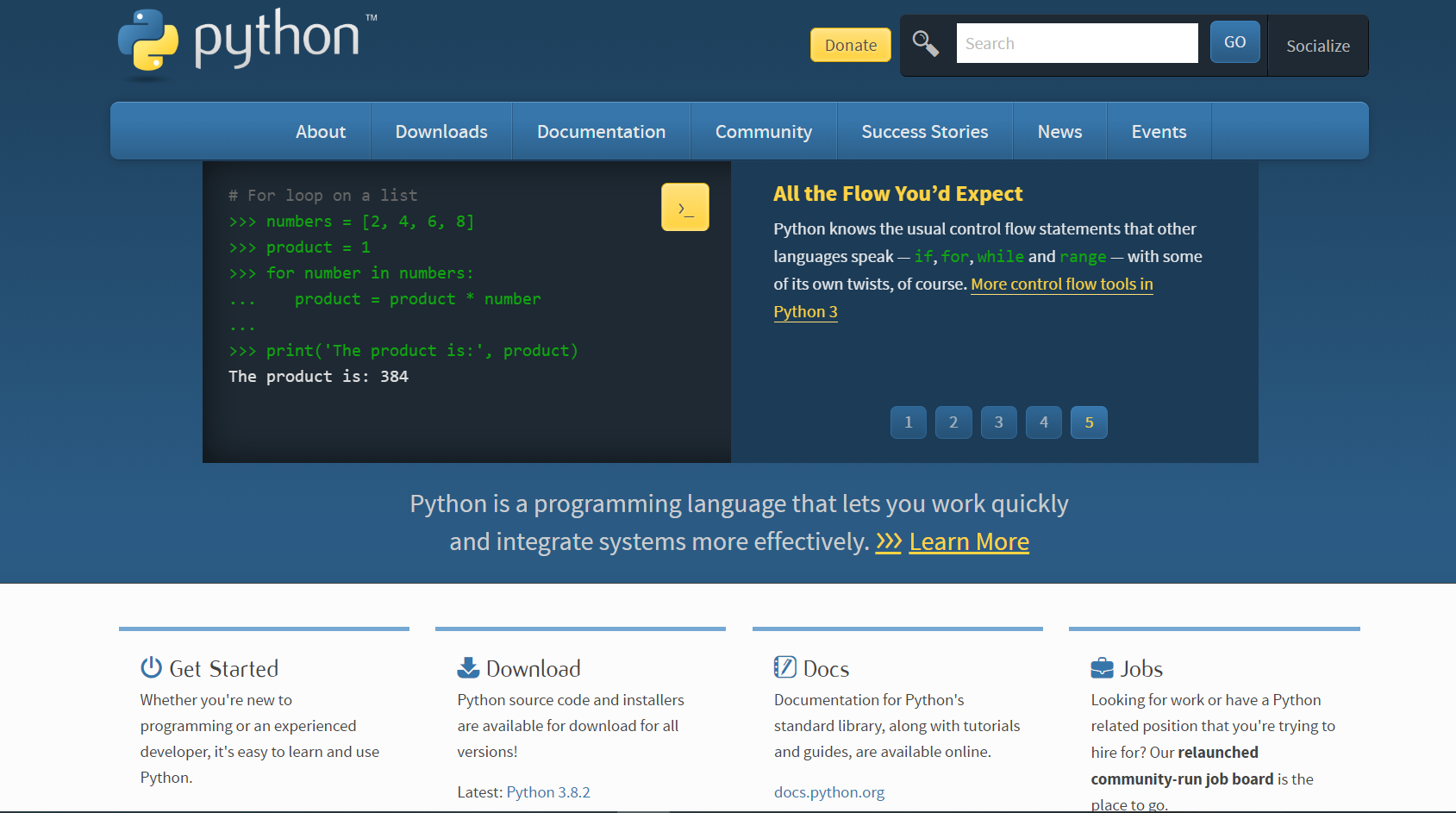
* Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
* Python has a simple syntax similar to the English language.
* Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
* Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
* Python can be treated in a procedural way, an object-orientated way or a functional way.

Python Syntax compared to other programming languages

* Python was designed for readability, and has some similarities to the English language with influence from mathematics.
* Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
* Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

How to Download Python IDLE?

Python IDLE can be downloaded for free from the website:<https://www.python.org/>



**HOW TO INSTALL PYTHON IN WINDOWS ?**

**Step 1: Download the Python Installer binaries**

1. Open the [official Python website](https://www.python.org/downloads/windows/) in your web browser. Navigate to the Downloads tab for Windows.
2. Choose the latest Python 3 release. In our example, we choose the latest Python 3.7.3 version.
3. Click on the link to download **Windows x86 executable installer** if you are using a 32-bit installer. In case your Windows installation is a 64-bit system, then download **Windows x86-64 executable installer**.



**Step 2: Run the Executable Installer**

1. Once the installer is downloaded, run the Python installer.
2. Check the **Install launcher for all users** check box. Further, you may check the **Add Python 3.7 to path** check box to include the interpreter in the execution path.



1. Select **Customize installation**.

Choose the optional features by checking the following check boxes:

* 1. Documentation
  2. [pip](https://www.journaldev.com/16160/python-pip)
  3. tcl/tk and IDLE (to install tkinter and IDLE)
  4. Python test suite (to install the standard library test suite of Python)
  5. Install the global launcher for `.py` files. This makes it easier to start Python
  6. Install for all users.



Click**Next**.

1. This takes you to **Advanced Options** available while installing Python. Here, select the **Install for all users** and **Add Python to environment variables** check boxes.

Optionally, you can select the **Associate files with Python**, **Create shortcuts for installed applications** and other advanced options. Make note of the python installation directory displayed in this step. You would need it for the next step.

After selecting the Advanced options, click **Install** to start installation.



1. Once the installation is over, you will see a **Python Setup Successful** window.



**Step 3: Add Python to environmental variables**

The last (optional) step in the installation process is to add Python Path to the System Environment variables. This step is done to access Python through the command line. In case you have added Python to environment variables while setting the Advanced options during the installation procedure, you can avoid this step. Else, this step is done manually as follows.

In the Start menu, search for “advanced system settings”. Select “View advanced system settings”. In the “System Properties” window, click on the “Advanced” tab and then click on the “Environment Variables” button.

Locate the Python installation directory on your system. If you followed the steps exactly as above, python will be installed in below locations:

* C:\Program Files (x86)\Python37-32: for 32-bit installation
* C:\Program Files\Python37-32: for 64-bit installation

The folder name may be different from “Python37-32” if you installed a different version. Look for a folder whose name starts with Python.

Append the following entries to PATH variable as shown below:





**Step 4: Verify the Python Installation**

You have now successfully installed Python 3.7.3 on Windows 10. You can verify if the Python installation is successful either through the command line or through the IDLE app that gets installed along with the installation.

Search for the command prompt and type “python”. You can see that Python 3.7.3 is successfully installed.



An alternate way to reach python is to search for “Python” in the start menu and clicking on IDLE (Python 3.7 64-bit). You can start coding in Python using the Integrated Development Environment(IDLE).



Hurray! You are ready to start developing Python applications in your Windows 10 system.

**PROGRAM 1: Armstrong Number**

n=int(input("enter the number.\n"))

n1=n

sum=0

while n!=0:

r=n%10

c=r\*r\*r

sum=sum+c

n=n//10

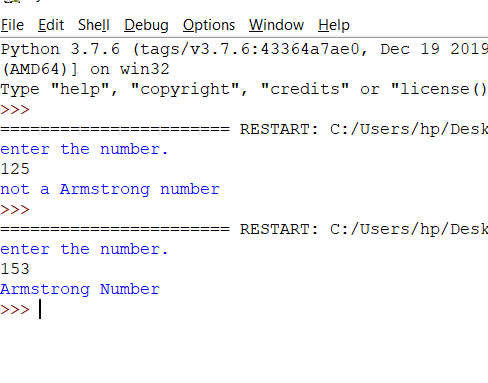
if sum==n1:

print("Armstrong Number")

else:

print("not a Armstrong number")

**OUTPUT:**



**PROGRAM 2: FACTORIAL**

n=int(input("enter the number.\n"))

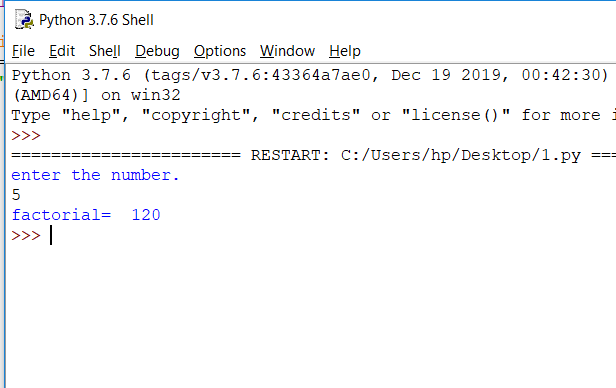
fact=1

for x in range(n,0,-1):

fact=fact\*x

print("factorial= ",fact)

**OUTPUT:**



**PROGRAM 3: FIBONACCI SERIES**

n=int(input("enter the number.\n"))

a,b=0,1

print("fibonacci series is: ")

print(a,end=" ")

print(b,end=" ")

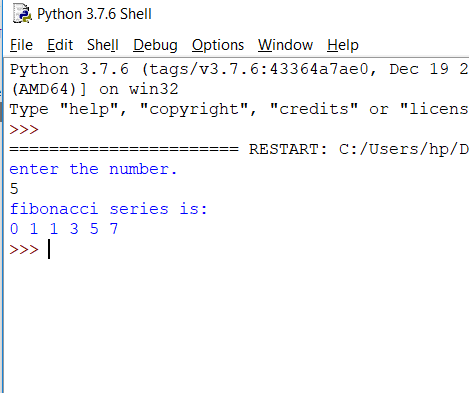
for x in range(2,n+1):

sum=a+b

a,b=b,x

print(sum,end=" ")

**OUTPUT:**



**PROGRAM 4: CHECK NUMBER IS EVEN OR ODD**

n=int(input("enter the no.\n"))

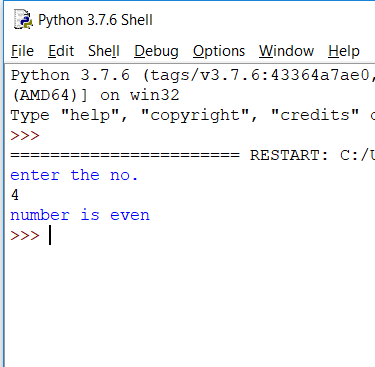
if n%2==0:

print("number is even")

else:

print("number is odd")

**OUTPUT:**



**PROGRAM 5:CALCULATE GROSS SALARY**

sal=int(input("enter the basic salary\n"))

if sal<=10000:

gs=sal+(sal/10)

elif 10000<sal<=20000:

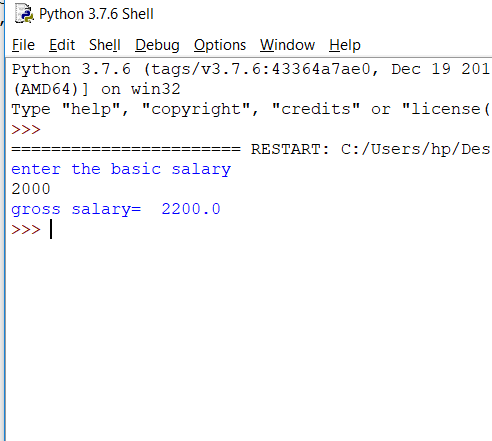
gs=sal+(sal/5)

else:

gs=sal+(sal\*0.3)

print("gross salary= ",gs)

**OUTPUT:**



**PROGRAM 6: CHECK IF LEAP YEAR**

n=int(input("enter the year\n"))

if n%4==0:

if n%100==0:

if n%400==0:

print("leap year")

else:

print("non leap year")

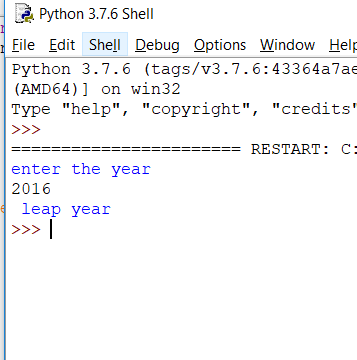
else:

print(" leap year")

else:

print("non leap year")

**OUTPUT:**



**PROGRAM 7:REVERSE OF A NUMBER**

n=int(input("enter the number.\n"))

sum=0

while n!=0:

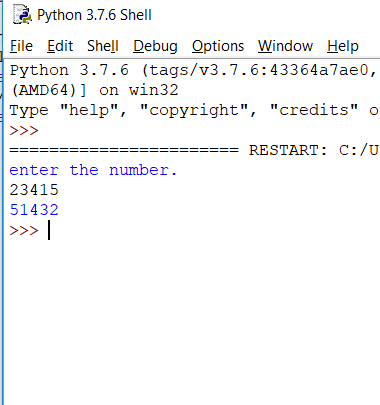
r=n%10

sum=sum\*10+r

n=n//10

print(sum)

**OUTPUT:**



**PROGRAM 8: SUM OF DIGITS OF A NUMBER**

n=int(input("enter the number.\n"))

sum=0

while n!=0:

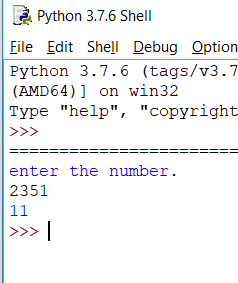
r=n%10

sum=sum+r

n=n//10

print(sum)

**OUTPUT:**



**PROGRAM 9: SWAPPING OF 2 NUMBERS USING THIRD VARIABLE**

x,y=int(input("enter the no.\n")),int(input("enter the no.\n"))

z=x

x=y

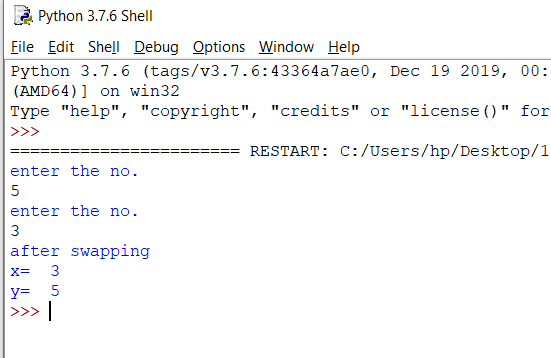
y=z

print("after swapping")

print("x= ",x)

print("y= ",y)

**OUTPUT:**



**PROGRAM 10: SWAPPING OF 2 NUMBERS WITHOUT USING THIRD VARIABLE**

x,y=int(input("enter the no.\n")),int(input("enter the no.\n"))

x=(x+y)

y=x-y

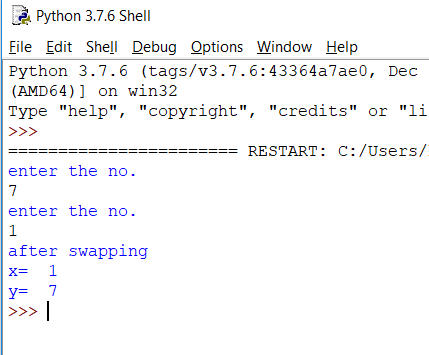
x=x-y

print("after swapping")

print("x= ",x)

print("y= ",y)

**OUTPUT:**



**PROGRAM 11:CHECK IF CHARACTER IS UPPERCASE OR LOWERCASE**

n=input("enter the character\n")

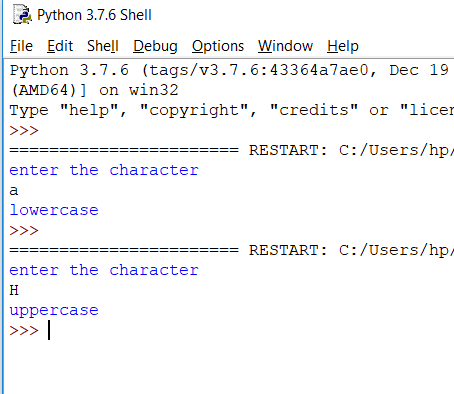
if 'A'<n<'Z':

print("uppercase")

else:

print("lowercase")

**OUTPUT:**



**PROGRAM 12: COUNT THE NUMBER OF CHARACTERS**

s=input("enter the word\n")

c=input("enter the char\n")

count=0

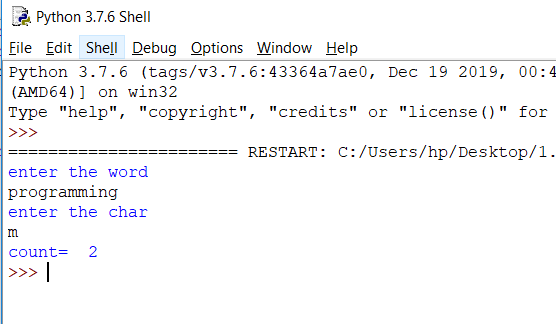
for x in s:

if x==c:

count+=1

print("count= ",count)

**OUTPUT:**



**PROGRAM 13:STRING WITH MAXIMUM LENGTH IN A LIST**

n=int(input("enter the number of word\n"))

l=[]

while n!=0:

s=input("enter the word\n")

l.append(s)

n=n-1

print(l)

max=len(l[0])

for x in range(len(l)):

f=len(l[x])

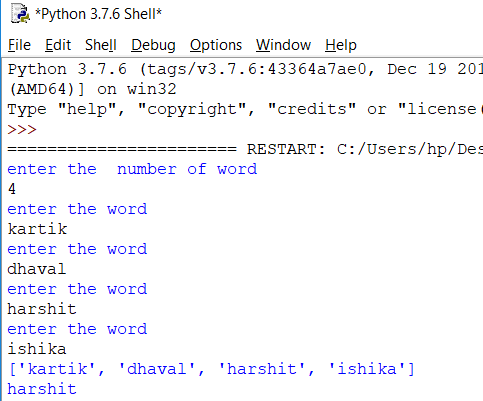
if max<f:

max=f

g=x

print(l[g])

**OUTPUT:**



**PROGRAM 14: OCCURENCES OF ALL CHARACTERS IN A STRING**

s=input("enter the word\n")

d = {}

for i in s:

if i in d:

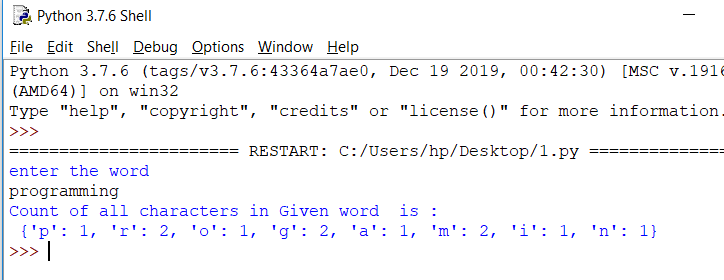
d[i] += 1

else:

d[i] = 1

print ("Count of all characters in Given word is :\n "+ str(d))

**OUTPUT:**



**PROGRAM 15: HAPPY NUMBER**

n=int(input("enter the number\n"))

def happyno(n):

sum,rem=0,0

while n>0:

rem=n%10

sum=sum+(rem\*rem)

n=n//10

return sum

p=n

while(p!=1 and p!=4):

p=happyno(p)

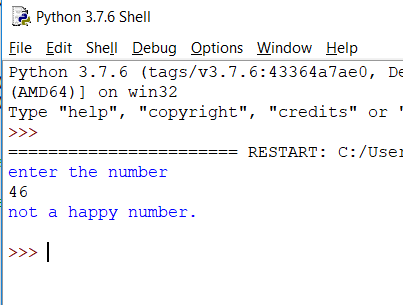
if p==1:

print("it is a happy number.\n")

if p==4:

print("not a happy number.\n")

**OUTPUT:**



**PROGRAM 16: PRINT THE STRING IN SORTED ORDER**

s=input("enter the string.\n")

l,l1,l2=[],[],[]

for x in s:

if x.isdigit():

l.append(x)

if x.isupper():

l1.append(x)

if x.islower():

l2.append(x)

print(l)

print(l1)

print(l2)

l=sorted(l)

l1=sorted(l1)

l2=sorted(l2)

print(l)

print(l1)

print(l2)

for x in l:

print(x,end="")

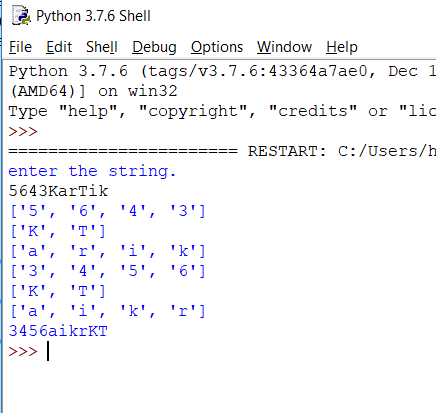
for x in l2:

print(x,end="")

for x in l1:

print(x,end="")

**OUTPUT:**



**PROGRAM 17:Declaration of Function (1):-**

def Sample():

print("Hello")

def add(x,y):

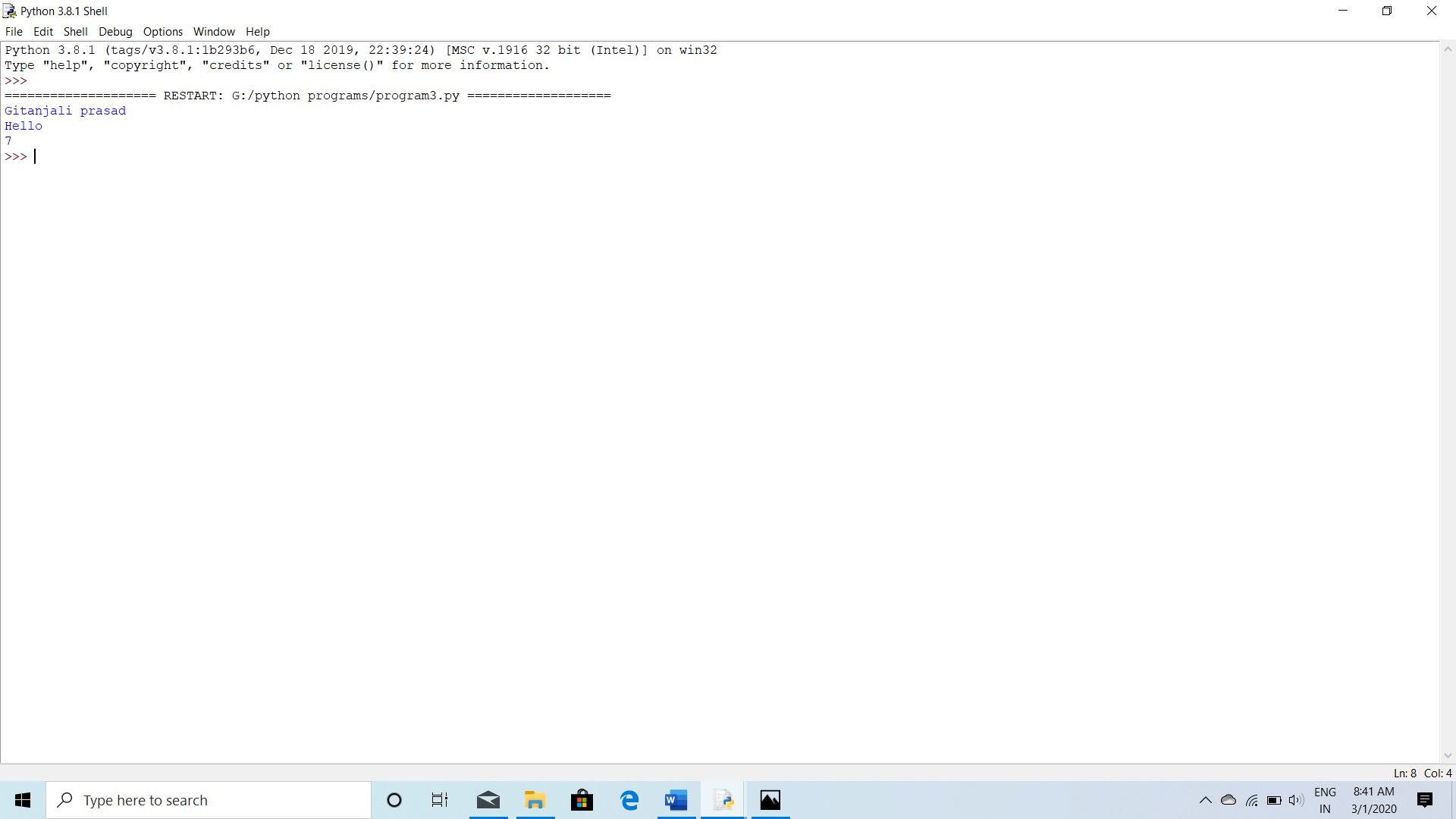
z=x+y

print(z)

Sample()

add(3,4)

**Output:**



**Program 18: declaration of Function (2):-**

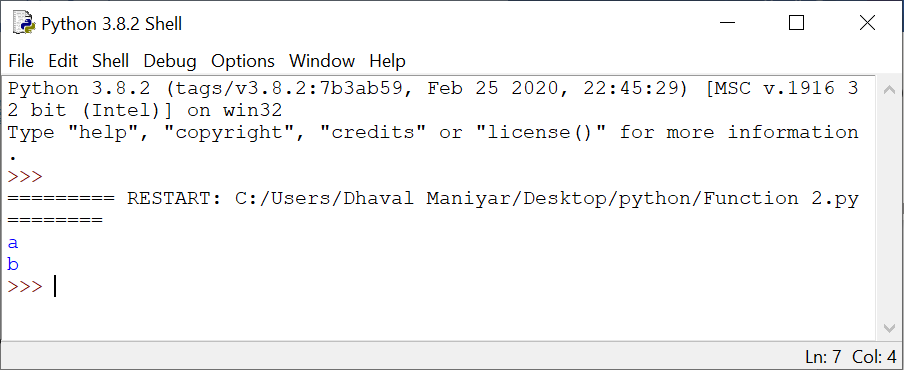
def test1(\*name):

print(name[0])

print(name[1])

test1('a','b','c')

**Output:**



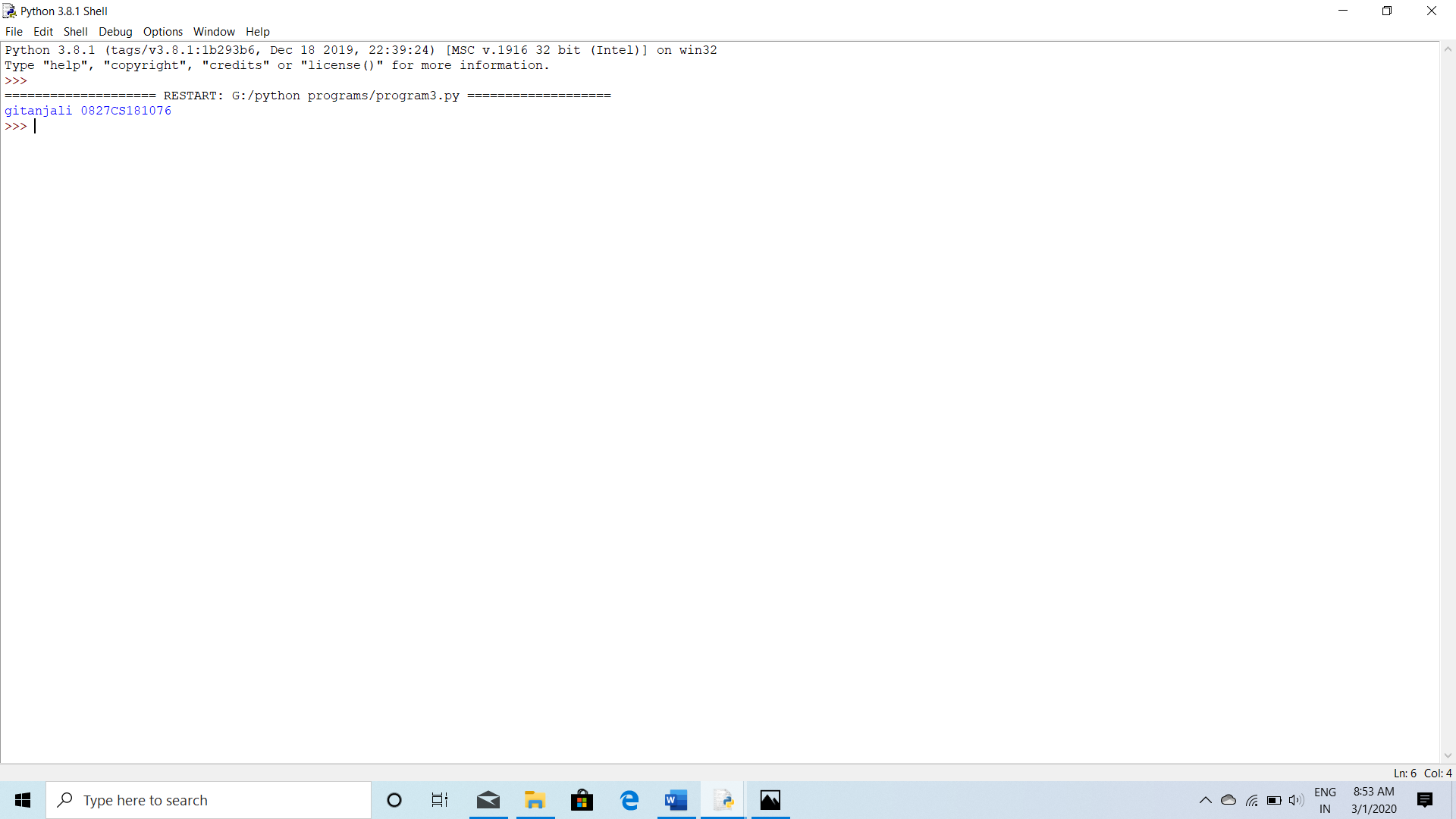
**Program 19:Declaration of Function (3):-**

def details(name,id):

print(name,id)

details(name="Kartik",id="0827CS181104")

**Output:**



**Program 20: Declaration of Function (4):-**

def details(city="Indore"):

print("I live in " ,city)

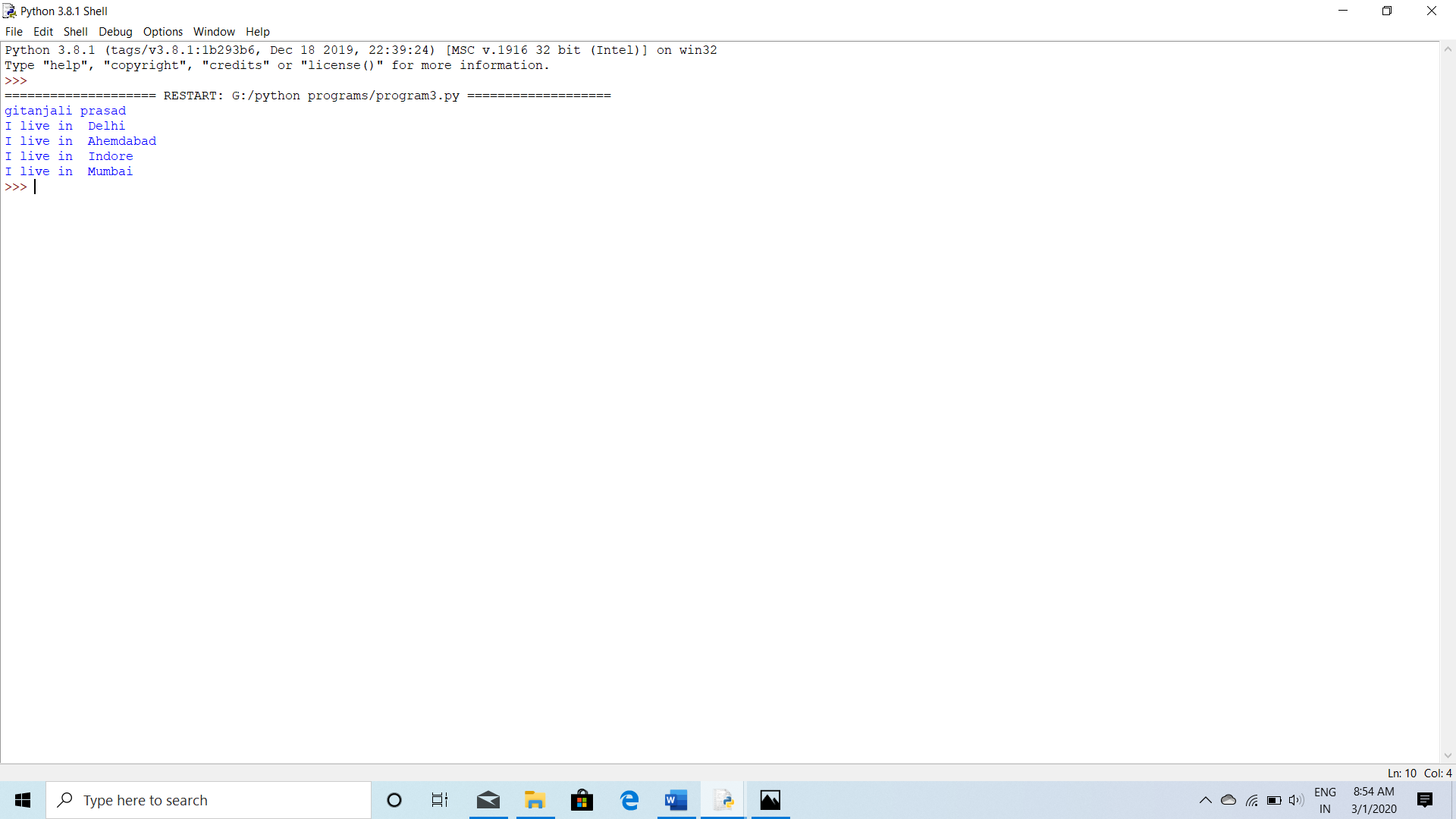
details("Delhi")

details("Ahemdabad")

details()

details("Mumbai")

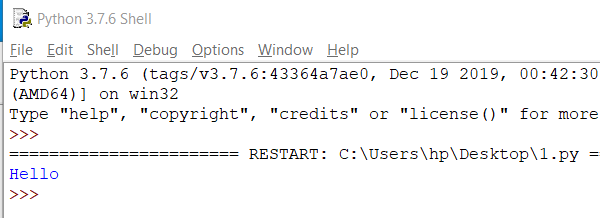
**Output:**



**PROGRAM 1.Write a python program to print"Hello".**

print("Hello")

**OUTPUT:**



**PROGRAM2.Write a python program to add,subtract,multiply and divide two numbers.**

num1=20

num2=10

summ=num1+num2

sub=num1-num2

multiply=num1\*num2

divide=num1//num2

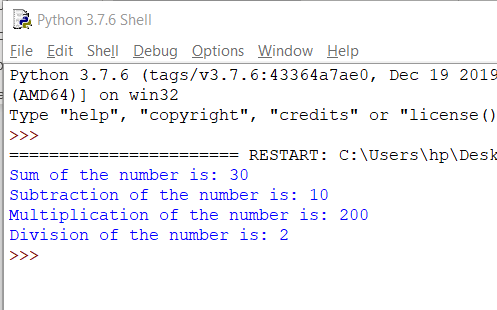
print("Sum of the number is:",summ)

print("Subtraction of the number is:",sub)

print("Multiplication of the number is:",multiply)

print("Division of the number is:",divide)

**OUTPUT:**



**PROGRAM 3.Write a python program to find percentage of a student.**

print("Marks obtained by student in each subject out of 100:")

total\_marks=0

hindi=90

english=80

maths=70

social\_science=60

science=50

print("Hindi:90")

print("English:80")

print("Maths:70")

print("Social Science:60")

print("Science:50")

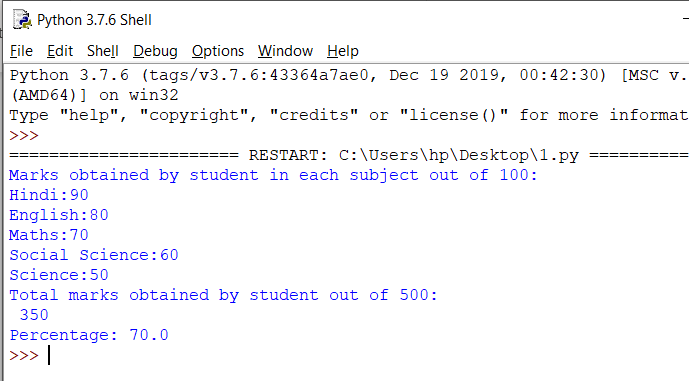
total\_marks+=hindi+english+maths+social\_science+science

print("Total marks obtained by student out of 500:\n",total\_marks)

per=total\_marks\*(100/500)

print("Percentage:",per)

**OUTPUT:**



**PROGRAM 4.Write a python program to swap two numbers using temporary variable.**

a=10

b=20

print("Before swapping")

print(a)

print(b)

c=a

a=b

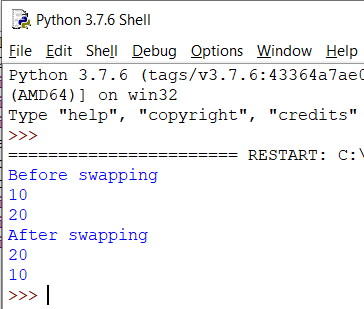
b=c

print("After swapping")

print(a)

print(b)

**OUTPUT:**



**PROGRAM 5.Write a python program to swap two numbers without using temporary variable.**

a=20

b=10

print("Before swapping")

print(a)

print(b)

a=a+b

b=a-b

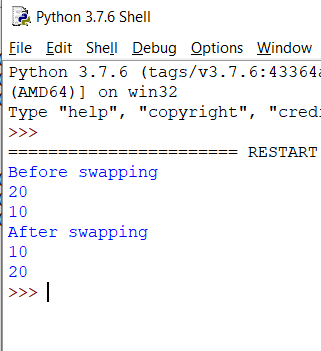
a=a-b

print("After swapping")

print(a)

print(b)

**OUTPUT:**



**PROGRAM 6.Write a python program to calculate factorial of a number**

num=5

fact=1

total=0

for i in range(0,5,1):

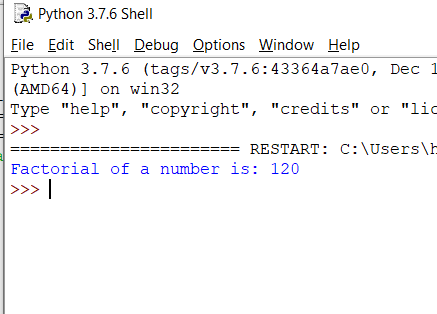
total=fact\*num

fact=total

num-=1

print("Factorial of a number is:",fact)

**OUTPUT:**



**PROGRAM 7.Write a python program to take user input.**

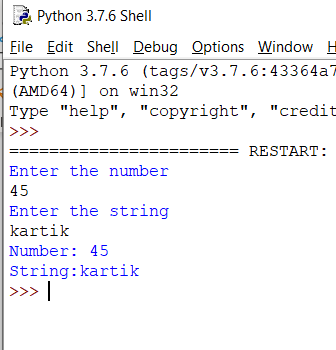
num=int(input("Enter the number\n"))

string=input("Enter the string\n")

print("Number:",num)

print("String:"+string)

**OUTPUT:**



**PROGRAM 8.Write a python program to accept two integers and check wheather they are equal or not.**

num1=int(input("Enter the first number "))

num2=int(input("Enter the second number "))

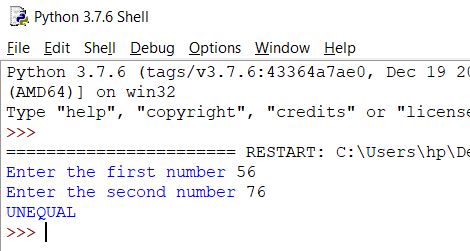
if num1==num2:

print("EQUAL")

else:

print("UNEQUAL")

**OUTPUT:**



**PROGRAM 9.Write a python program to find even and odd.**

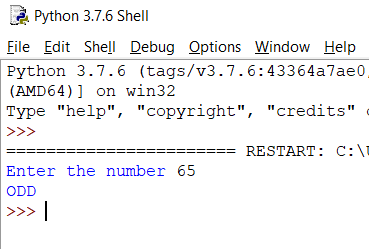
num=int(input("Enter the number "))

if num%2==0:

print("EVEN")

else:print("ODD")

**OUTPUT:**



**PROGRAM 10.Write a python program to find greatest among three numbers.**

num1=int(input("Enter the first number "))

num2=int(input("Enter the second number "))

num3=int(input("Enter the third number "))

if num1>num2 and num1>num3:

print("First number is Greater")

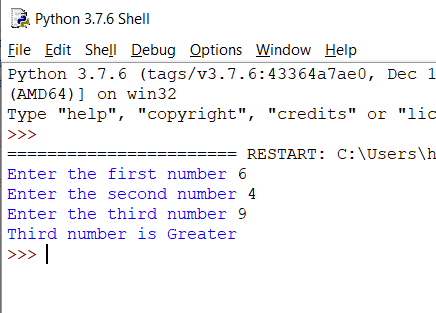
elif num2>num1 and num2>num3:

print("Second number is Greater")

else:

print("Third number is Greater")

**OUTPUT:**



**PROGRAM 11.Write a python program to check whether a given number is positive or negative.**

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

if num>0:

print("Positive")

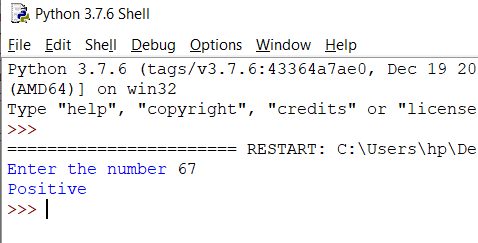
elif num<0:

print("Negative")

else:

print("Neutral")

**OUTPUT:**



**PROGRAM 12.Write a python program to check whether entered character is vowel or consonant**.

char=input("Enter the character ")

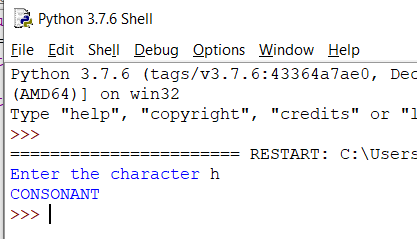
if char=='a' or char=='e' or char=='i' or char=='o' or char=='u' or char=='A' or char=='E' or char=='I' or char=='O' or char=='U':

print("VOWEL")

else:

print("CONSONANT")

**OUTPUT:**



**PROGRAM 13.Write a python program to check result of student(pass or fail).**

total\_marks=0

hindi=float(input("Enter the marks of Hindi out of 100:"))

english=float(input("Enter the marks of English out of 100:"))

maths=float(input("Enter the marks of Maths out of 100:"))

social\_science=float(input("Enter the marks of Social Science out of 100:"))

science=float(input("Enter the marks of Science out of 100:"))

total\_marks+=hindi+english+maths+social\_science+science

per=total\_marks\*(100/500)

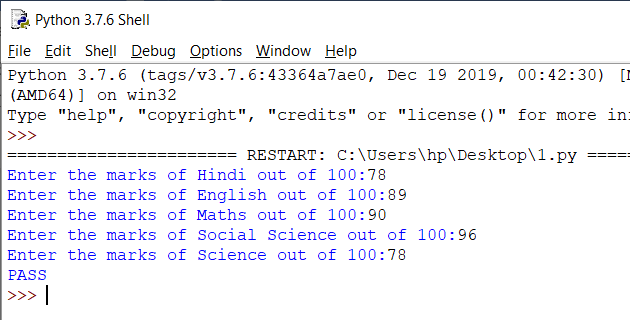
if per>0 and per<=33:

print("FAIL")

elif per>33 and per<=100:

print("PASS")

**OUTPUT:**



**PROGRAM 14.Write a python program to find a given year is leap or not.**

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

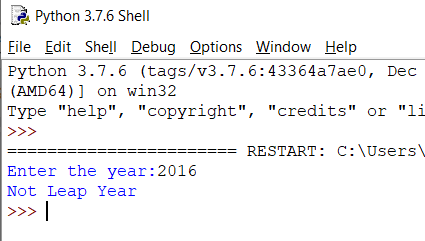
if year%4==0 and year%100==0 and year%400==0:

print("Leap Year")

else:

print("Not Leap Year")

**OUTPUT:**

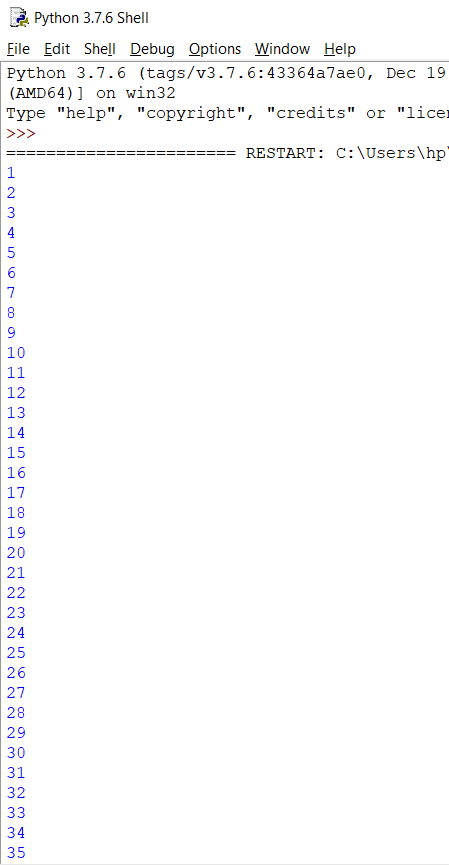


**PROGRAM 15.Write a python program to display numbers from 1 to 100 using loop.**

for i in range(1,101,1):

print(i)

**OUTPUT:**



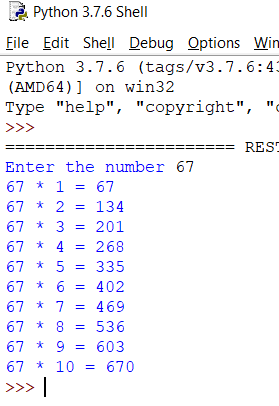
**PROGRAM 16.Write a python program to display table of a number taken from user input.**

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

for i in range(1,11,1):

print(num ,"\*", i,"=",num\*i )

**OUTPUT:**



**PROGRAM 17.Write a python program to print factorial of a num received as a user input.**

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

fact=1

total=0

for i in range(0,num,1):

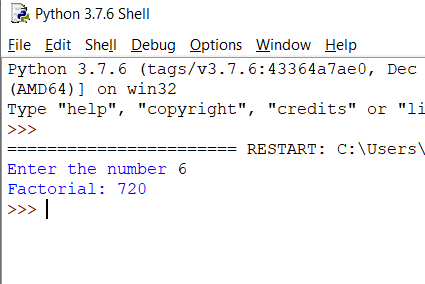
total=fact\*num

fact=total

num-=1

print("Factorial:",fact)

**OUTPUT:**



**PROGRAM 18.Write a python program to find fibonacci series upto n terms**.

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

a=0

b=1

print(a,end=" ")

print(b,end=" ")

for i in range(0,num,1):

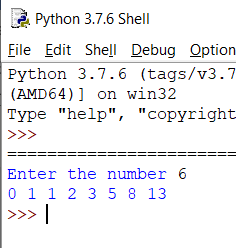
c=a+b

print(c,end=" ")

a=b

b=c

**OUTPUT:**



**PROGRAM 19.Write a python program to generate prime number from 1 to 50**.

for num in range(1,51):

count=0

for i in range(2,(num//2+1)):

if (num%i==0):

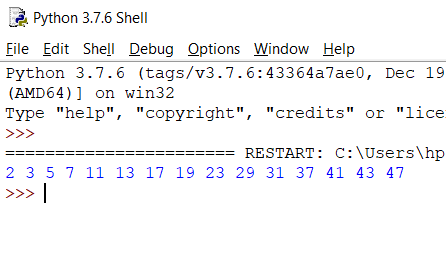
count+=1

break

if (count==0 and num!=1):

print(num,end=" ")

**OUTPUT:**



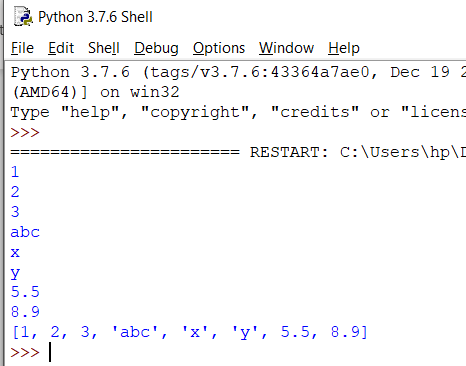
**PROGRAM 20.Write a python to create a list**.

l=[1,2,3,"abc","x","y",5.5,8.9]

for i in l:

print(i)

**OUTPUT:**



**PROGRAM 21.Write a python program to access elements of list.**

l=[1,2,3,"abc","x","y",5.5,8.9]

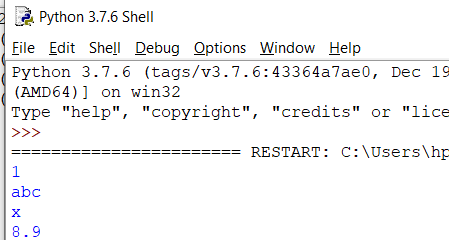
print(l[0])

print(l[3])

print(l[4]) #accessing elements of list

print(l[7])

**OUTPUT:**



**PROGRAM 22.Write a python program to print even numbers present in list.**

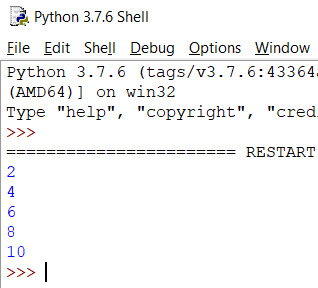
l=[1,2,3,4,5,6,7,8,9,10]

for i in l:

if i%2==0:

print(i)

**OUTPUT:**



**PROGRAM 23.Write a python program to multiplies all the items in a list**

l=[1,2,3,4,5,6]

num=1

total=0

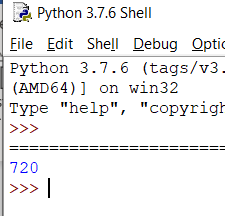
for i in l:

total=num\*i

num=total

print(total)

**OUTPUT:**



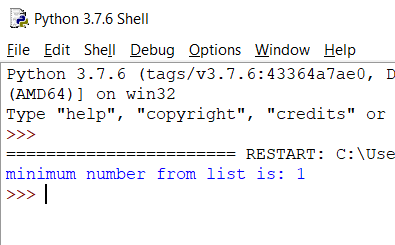
**PROGRAM 24.Write a python program to get the smallest number from list**.

l=[2,4,6,1,9,7]

min\_num=min(l)

print("minimum number from list is:",min\_num)

**OUTPUT:**



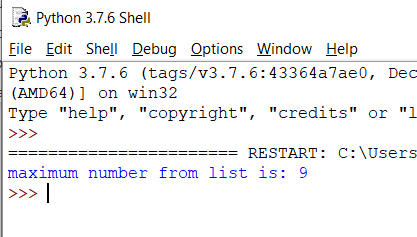
**PROGRAM 25.Write a python program to get the largest number from a list**.

l=[2,4,6,1,9,7]

max\_num=max(l)

print("maximum number from list is:",max\_num)

**OUTPUT:**

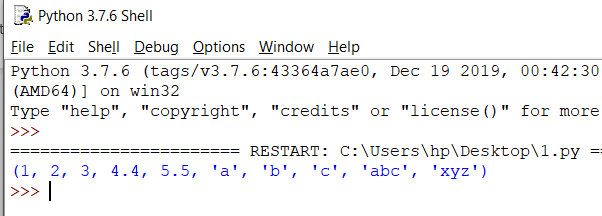


**PROGRAM 26.Write a python program to create a tuple**.

t=(1,2,3,4.4,5.5,'a','b','c','abc','xyz')

print(t)

**OUTPUT:**

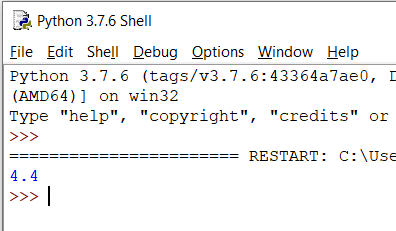


**PROGRAM 27.Write a python program to create a tuple with numbers and print one item.**

t=(1,2,3,4.4,5.5,'a','b','c','abc','xyz')

print(t[3])

**OUTPUT:**



**PROGRAM 28.Write a python program to add an item in a tuple.**

t=(1,2,3,4.4,5.5,'a','b','c','abc','xyz')

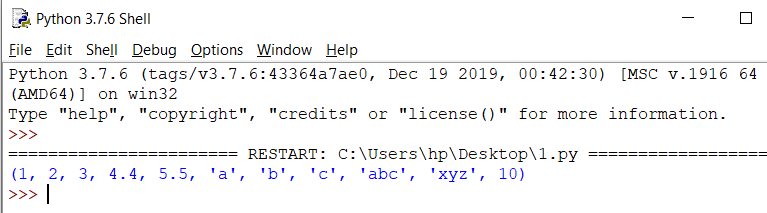
l=list(t)

l.append(10)

t=tuple(l)

print(t)

**OUTPUT:**



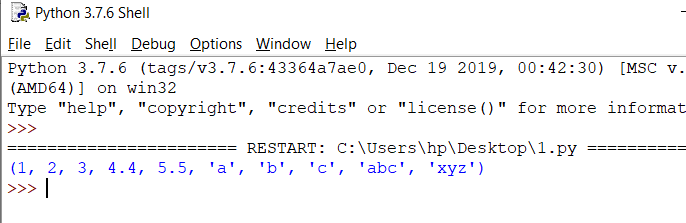
**PROGRAM 29.Write a python program to convert a list to a tuple**.

l=[1,2,3,4.4,5.5,'a','b','c','abc','xyz']

t=tuple(l)

print(t)

**OUTPUT:**



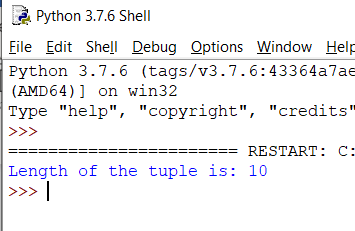
**PROGRAM 30.Write a python program to find the length of a tuple.**

t=(1,2,3,4.4,5.5,'a','b','c','abc','xyz')

length=len(t)

print("Length of the tuple is:",length)

**OUTPUT:**

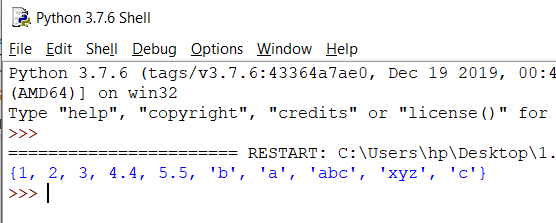


**PROGRAM 31.Write a python program to create a set**.

s={1,2,3,4.4,5.5,'a','b','c','abc','xyz'}

print(s)

**OUTPUT:**



**PROGRAM 32.Write a python program to add member(s) in a set.**

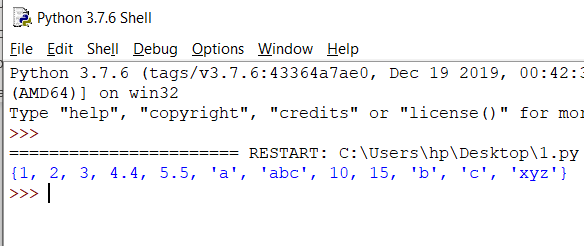
s={1,2,3,4.4,5.5,'a','b','c','abc','xyz'}

s.add(10)

s.add(15)

print(s)

**OUTPUT:**



**PROGRAM 33.Write a python program to remove item(s) from set.**

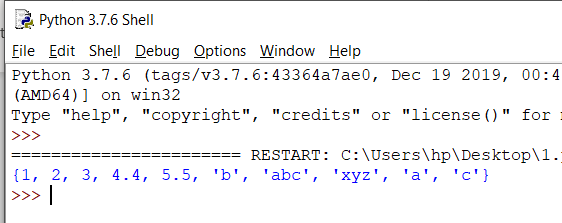
s={1,2,3,4.4,5.5,'a','b','c',10,15,'abc','xyz'}

s.remove(10)

s.remove(15)

print(s)

**OUTPUT:**



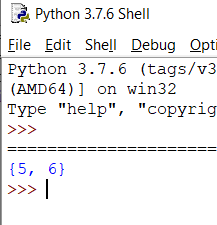
**PROGRAM 34.Write a python program to create an intersection of sets.**

s1={1,2,3,4,5,6}

s2={5,6,7,8,9,10}

intersection\_set=s1&s2

print(intersection\_set)



**PROGRAM 35.Write a python program to create an union of sets**.

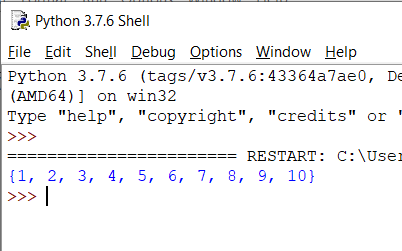
s1={1,2,3,4,5,6}

s2={5,6,7,8,9,10}

union\_set=s1|s2

print(union\_set)

**OUTPUT:**



**PROGRAM 36.Write a python program to create set difference.**

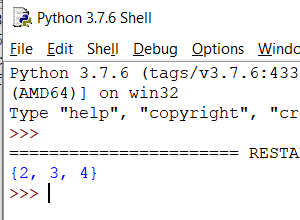
s1={1,2,3,4,5,6}

s2={1,5,6}

difference\_set=s1-s2

print(difference\_set)

**OUTPUT:**



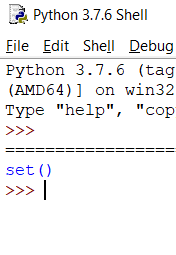
**PROGRAM 37.Write a python program to clear a set.**

s={1,2,3,4,5,6}

s.clear()

print(s)

**OUTPUT:**

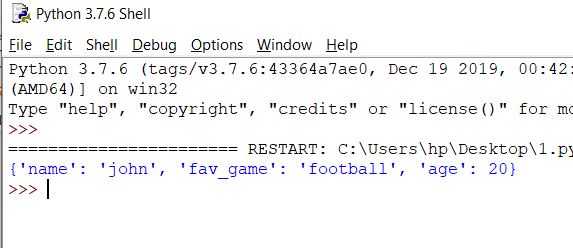


**PROGRAM 38.Write a python program to create dictionary.**

d={'name':'john','fav\_game':'football','age':20}

print(d)

**OUTPUT:**



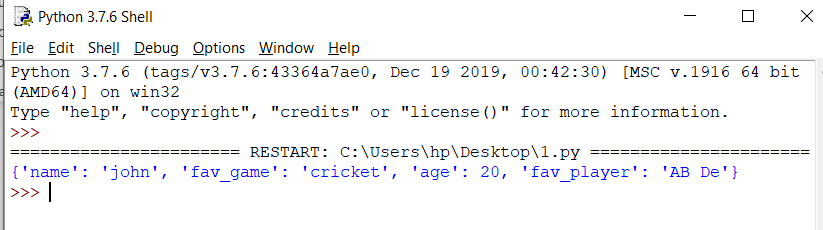
**PROGRAM 39.Write a python program to add a key to a dictionary.**

d={'name':'john','fav\_game':'cricket','age':20}

d['fav\_player']='AB De' #adding new key to dict

print(d)

**OUTPUT:**



**PROGRAM 40.Write a python program to check whether a given key already exist in a dictionary.**

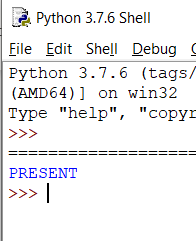
d={'name':'john','fav\_game':'cricket','age':20,'fav\_player':'AB de'}

if 'fav\_game' in d:

print("PRESENT")

else:

print("NOT PRESENT")



**PROGRAM 41.Write a python program to merge two python dictionaries.**

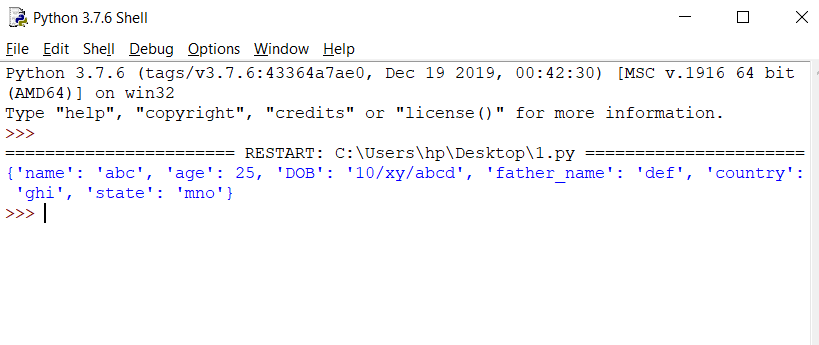
user\_info1={'name':'abc','age':25,'DOB':"10/xy/abcd"}

user\_info2={'father\_name':'def','country':'ghi','state':'mno'}

user\_info1.update(user\_info2)

print(user\_info1)

**OUTPUT:**



**PROGRAM 42.Write a python program to sum all the items in a dictionary.**

d={'a':10,'b':20,'c':30,'d':40,'e':50}

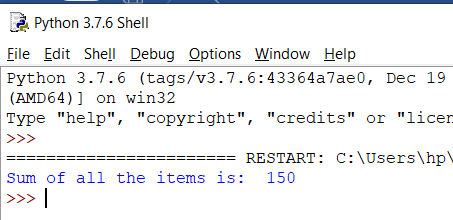
total=0

for i in d:

total+=d[i]

print("Sum of all the items is: ",total)

**OUTPUT:**



**PROGRAM 43.Write a python function to find average of five numbers passed parameters.**

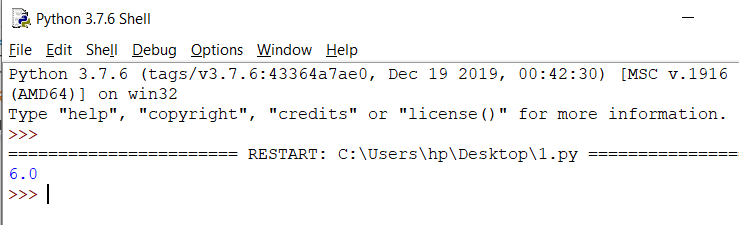
def find\_average(a,b,c,d,e):

average=(a+b+c+d+e)/5

return average

print(find\_average(2,4,6,8,10))

**OUTPUT:**



**PROGRAM 44.Write a python function to take list as argument and remove odd numbers from list.**

def list\_func(list):

for i in list:

if i%2!=0:

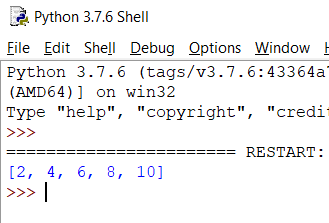
list.remove(i)

print(list)

list=[1,2,3,4,5,6,7,8,9,10]

list\_func(list)

**OUTPUT:**



**PROGRAM 45.Write a python function to find fibonacci series upto to n terms.**

def fibonacci(num,a,b):

for i in range(0,num,1):

c=a+b

print(c,end=" ")

a=b

b=c

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

a=0

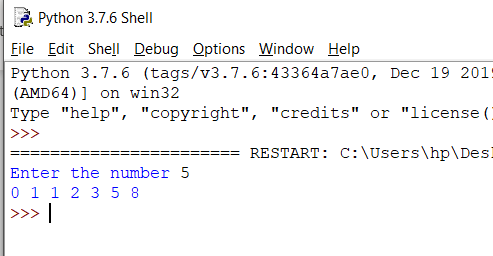
b=1

print(a,end=" ")

print(b,end=" ")

fibonacci(num,a,b)

**OUTPUT:**



**PROGRAM 46.Write a python function to find factorial of a number.**

def factorial(num):

fact=1

total=0

for i in range(0,5,1):

total=fact\*num

fact=total

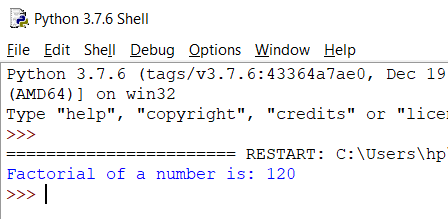
num-=1

print("Factorial of a number is:",fact)

num=5

factorial(num)

**OUTPUT:**



**PROGRAM 47. Write a python function to generate a table of any number entered by user.**

def table(num):

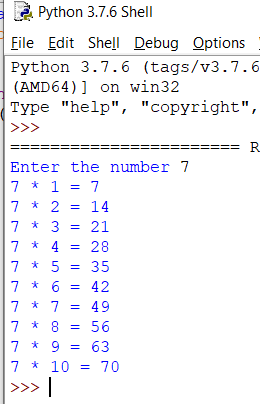
for i in range(1,11,1):

print(num ,"\*", i,"=",num\*i )

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

table(num)

**OUTPUT:**



**PROGRAM 48. Write a program to check whether given number is prime or composite.**

def prime(num):

if num > 1:

for i in range(2, num//2):

if (num % i) == 0:

c= 0

break

else:

c= 1

else:

c= 0

if c==0:

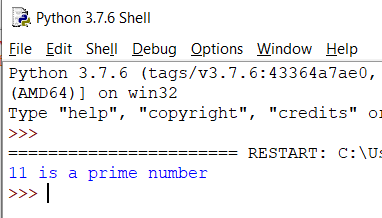
print(num, "is a composite number")

else:

print(num, "is a prime number")

prime(11)

**OUTPUT:**



**PROGRAM 50. Write a python function to check whether input number is even or odd.**

def even\_odd(num):

if num%2==0:

print("EVEN")

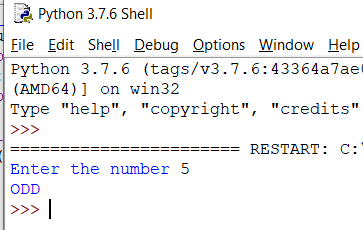
else:

print("ODD")

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

even\_odd(num)

**OUTPUT:**



**PROGRAM 51.Write a python function to sum all the numbers in a list**.

def sum\_all(list):

total=0

for i in list:

total+=i

print("Sum of all the list elements is:",total)

list=[1,2,3,4,5,6,7,8,9,10]

sum\_all(list)

**OUTPUT:**

