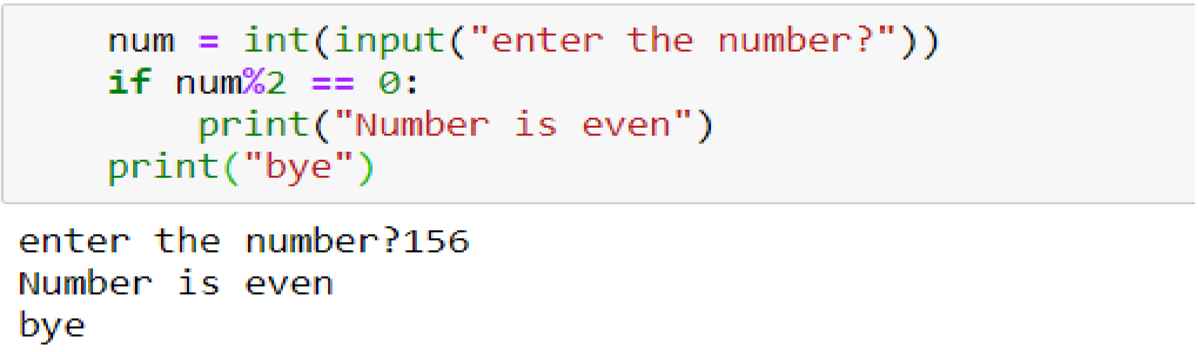
A.If else statements

Objective: To perform if-else loop

Outcome: check whether number is even or odd, print grade, vote eligibility using if else statements.

Conclusion: Successfully implemented if-else loop.

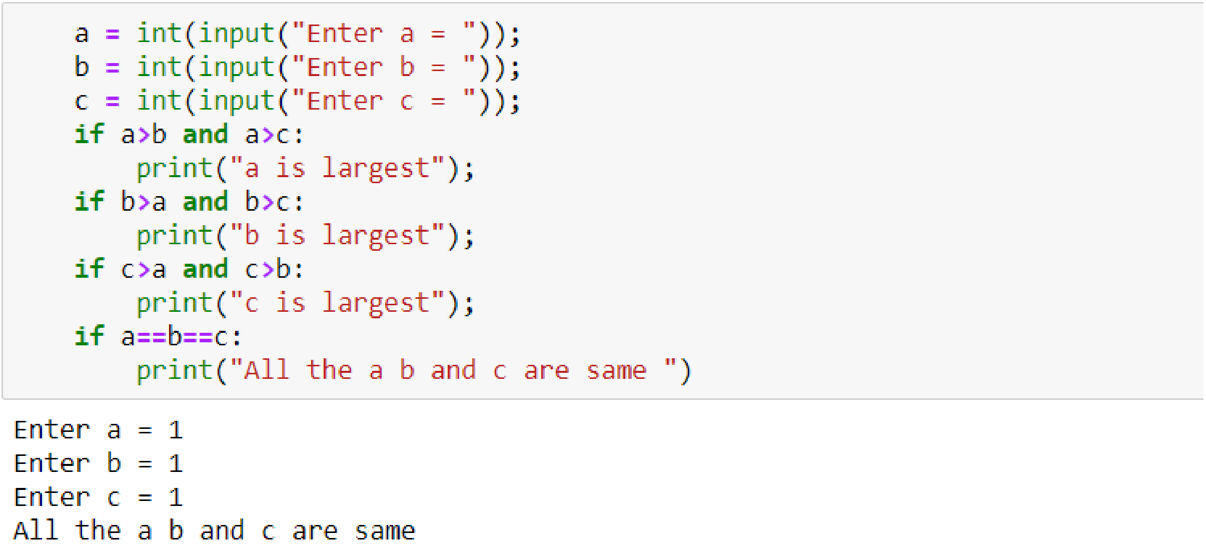
1.Program to check if the number is even



```
num = int(input("enter the number?"))
if num%2 == 0:
    print("Number is even")
print("bye")
```

```
enter the number?156
Number is even
bye
```

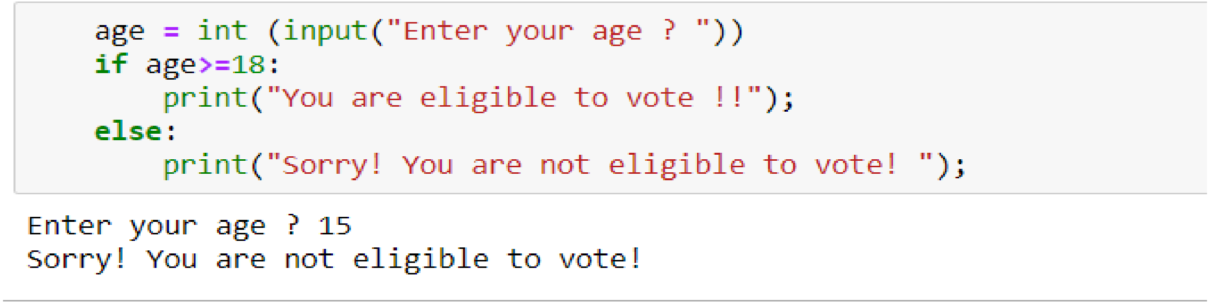
2.Program to check the largest among three numbers



```
a = int(input("Enter a = "));
b = int(input("Enter b = "));
c = int(input("Enter c = "));
if a>b and a>c:
    print("a is largest");
if b>a and b>c:
    print("b is largest");
if c>a and c>b:
    print("c is largest");
if a==b==c:
    print("All the a b and c are same ")
```

```
Enter a = 1
Enter b = 1
Enter c = 1
All the a b and c are same
```

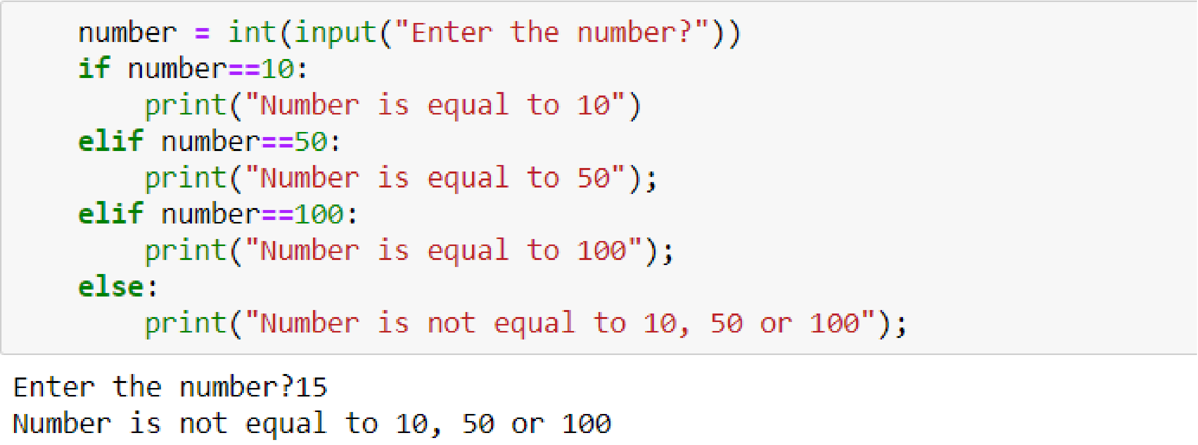
3.Program to evaluate the age eligible for voting



```
age = int (input("Enter your age ? "))
if age>=18:
    print("You are eligible to vote !!");
else:
    print("Sorry! You are not eligible to vote! ");
```

```
Enter your age ? 15
Sorry! You are not eligible to vote!
```

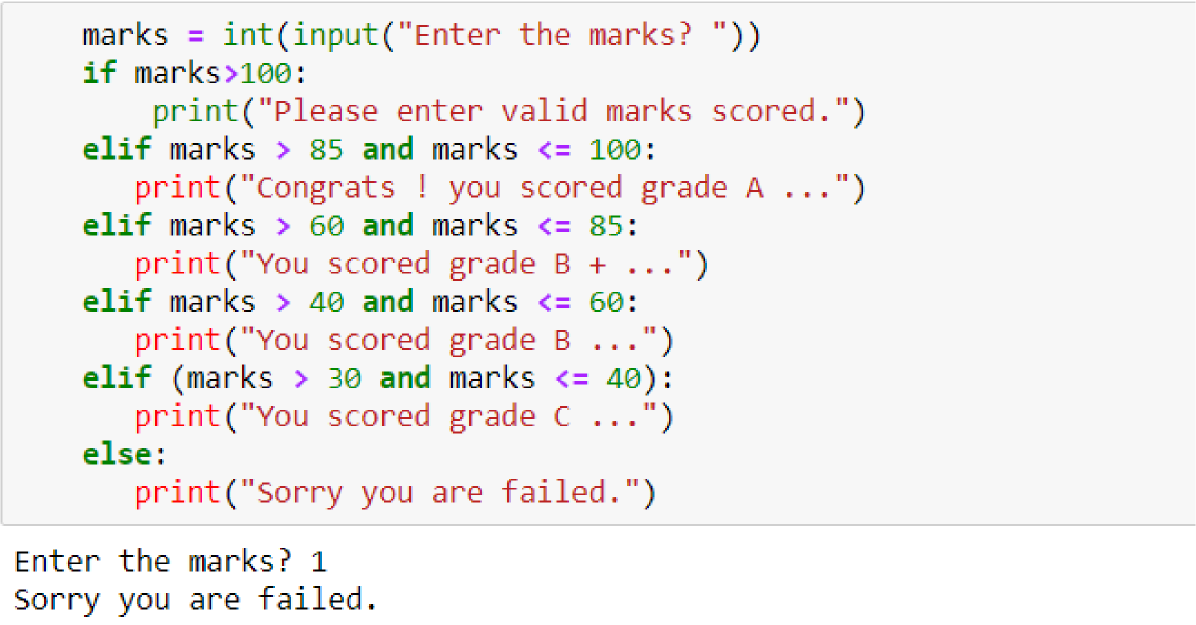
4.Program to check whether the number is equal to 10,50 or 100



```
number = int(input("Enter the number?"))
if number==10:
    print("Number is equal to 10")
elif number==50:
    print("Number is equal to 50");
elif number==100:
    print("Number is equal to 100");
else:
    print("Number is not equal to 10, 50 or 100");
```

```
Enter the number?15
Number is not equal to 10, 50 or 100
```

5.Program to calculate the grade of a student



```
marks = int(input("Enter the marks? "))
if marks>100:
     print("Please enter valid marks scored.")
elif marks > 85 and marks <= 100:
    print("Congrats ! you scored grade A ...")
elif marks > 60 and marks <= 85:
    print("You scored grade B + ...")
elif marks > 40 and marks <= 60:
    print("You scored grade B ...")
elif (marks > 30 and marks <= 40):
    print("You scored grade C ...")
else:
    print("Sorry you are failed.")
```

```
Enter the marks? 1
Sorry you are failed.
```

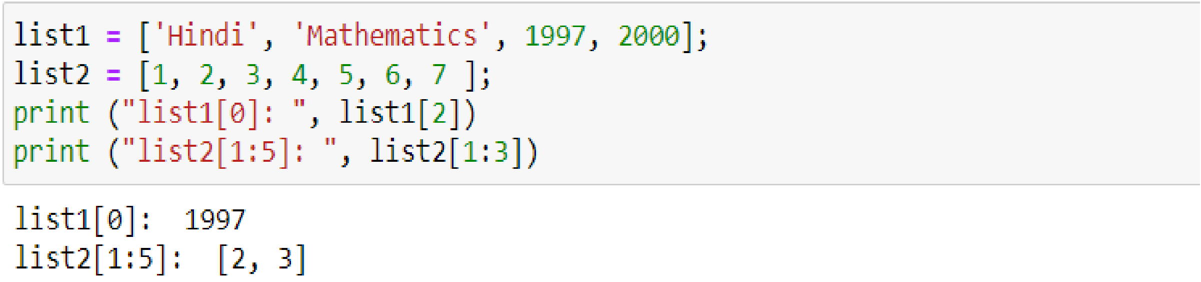
B.List

Objective: To perform lists and dictionary operations.

Outcome: Tuple operations like Concatenation, Adding/Deleting elements, accessing elements, replacing elements ,Appending lists

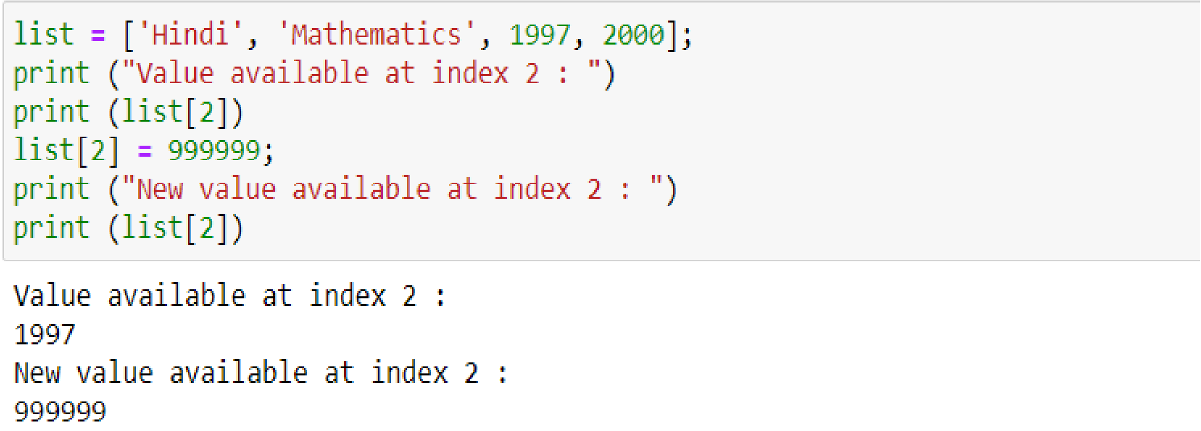
Conclusion: Successfully completed the Dictionaries and tuple operations.

1.Program to implement the list and manipulate.



```
list1 = ['Hindi', 'Mathematics', 1997, 2000];
list2 = [1, 2, 3, 4, 5, 6, 7 ];
print ("list1[0]: ", list1[2])
print ("list2[1:5]: ", list2[1:3])
```

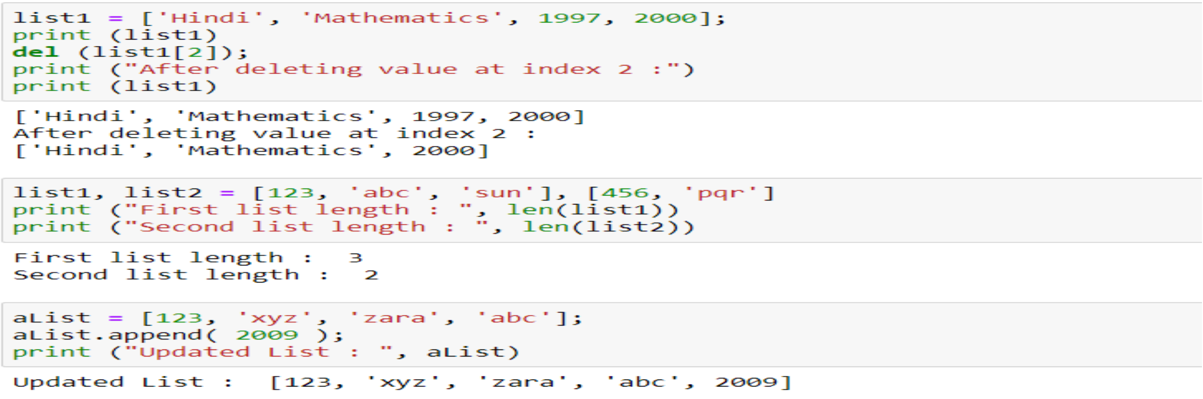
```
list1[0]:  1997
list2[1:5]:  [2, 3]
```



```
list = ['Hindi', 'Mathematics', 1997, 2000];
print ("Value available at index 2 : ")
print (list[2])
list[2] = 999999;
print ("New value available at index 2 : ")
print (list[2])
```

```
Value available at index 2 :
1997
New value available at index 2 :
999999
```

2.Program to implement the append function on list.



```
list1 = ['Hindi', 'Mathematics', 1997, 2000];
print (list1)
del (list1[2]);
print ("After deleting value at index 2 :")
print (list1)
```

```
['Hindi', 'Mathematics', 1997, 2000]
After deleting value at index 2 :
['Hindi', 'Mathematics', 2000]
```

```
list1, list2 = [123, 'abc', 'sun'], [456, 'pqr']
print ("First list length : ", len(list1))
print ("Second list length : ", len(list2))
```

```
First list length :  3
Second list length :  2
```

```
aList = [123, 'xyz', 'zara', 'abc'];
aList.append( 2009 );
print ("Updated List : ", aList)
```

```
Updated List :  [123, 'xyz', 'zara', 'abc', 2009]
```

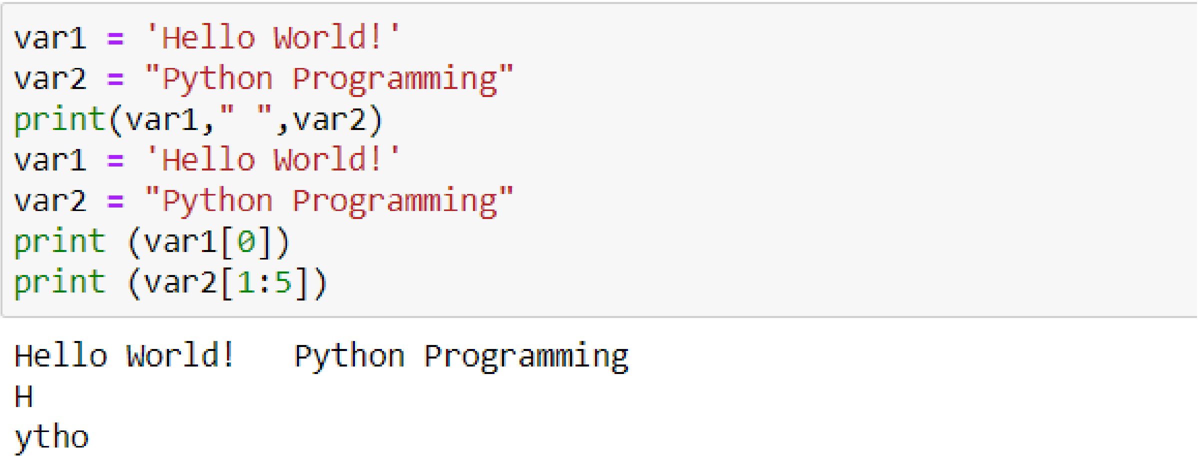
C.String operations

Objective: To perform String operations using break, continue

Outcome: String operations like Concatenation, Append,

Conclusion: Successfully implemented String operations

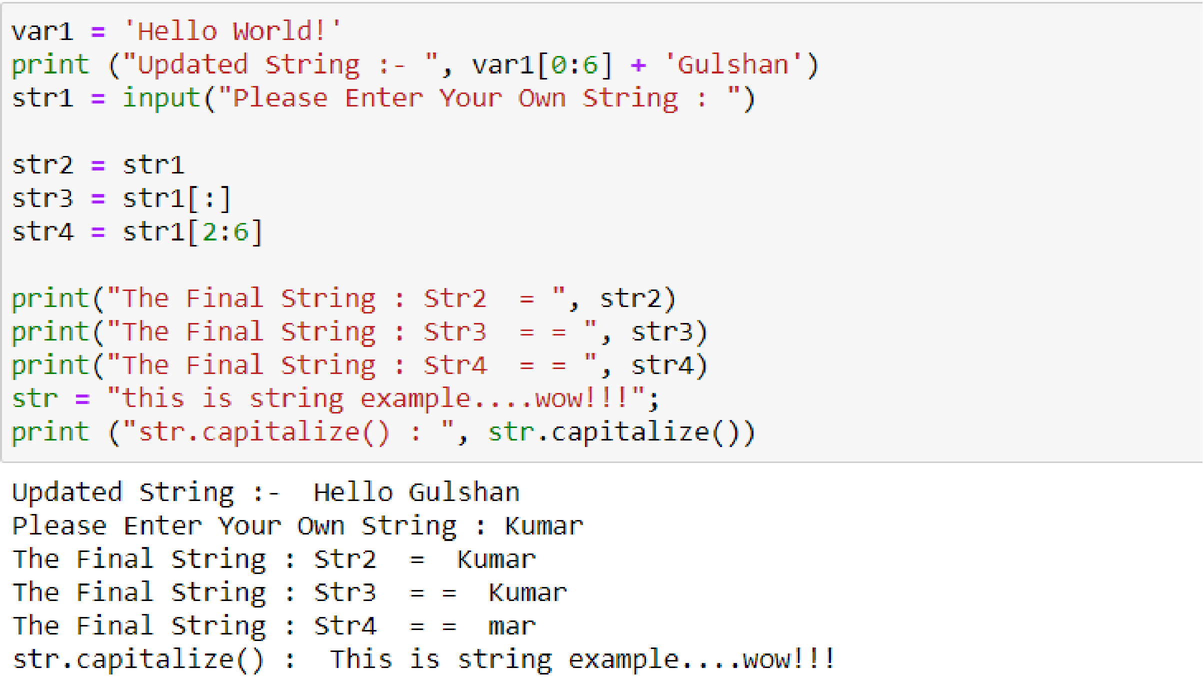
1.Program to access the character from the string.



```
var1 = 'Hello World!'
var2 = "Python Programming"
print(var1," ",var2)
var1 = 'Hello World!'
var2 = "Python Programming"
print (var1[0])
print (var2[1:5])
```

```
Hello World!   Python Programming
H
ytho
```

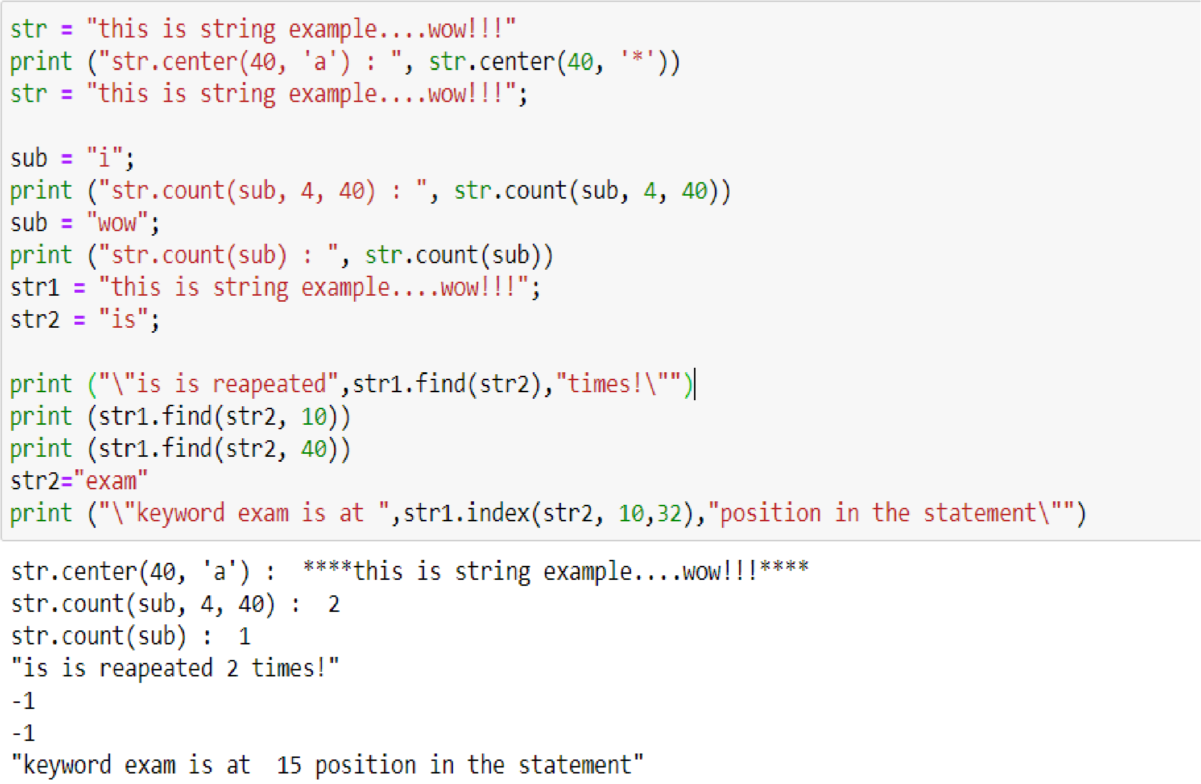
2. program to input string and access the string and capitalize the string



```
var1 = 'Hello World!'
print ("Updated String :- ", var1[0:6] + 'Gulshan')
str1 = input("Please Enter Your Own String : ")

str2 = str1
str3 = str1[:]
str4 = str1[2:6]

print("The Final String : Str2  = ", str2)
print("The Final String : Str3  = = ", str3)
print("The Final String : Str4  = = ", str4)
str = "this is string example....wow!!!";
print ("str.capitalize() : ", str.capitalize())
```

```
Updated String :-  Hello Gulshan
Please Enter Your Own String : Kumar
The Final String : Str2  =  Kumar
The Final String : Str3  = =  Kumar
The Final String : Str4  = =  mar
str.capitalize() :  This is string example....wow!!!
```

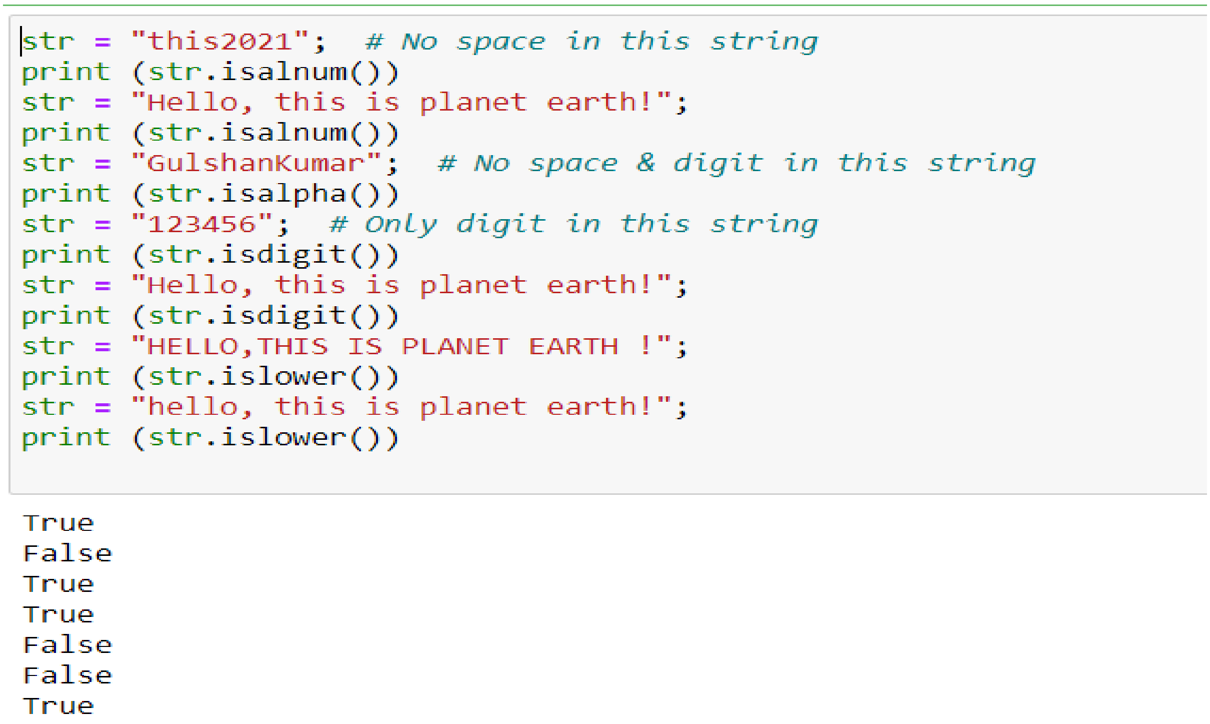
3.program to show the function of count()-returns the number of occurrences of substring sub in the range [start, end].



```
str = "this is string example....wow!!!"
print ("str.center(40, 'a') : ", str.center(40, '*'))
str = "this is string example....wow!!!";

sub = "i";
print ("str.count(sub, 4, 40) : ", str.count(sub, 4, 40))
sub = "wow";
print ("str.count(sub) : ", str.count(sub))
str1 = "this is string example....wow!!!";
str2 = "is";

print ("\"is is reapeated",str1.find(str2),"times!\"")
print (str1.find(str2, 10))
print (str1.find(str2, 40))
str2="exam"
print ("\"keyword exam is at ",str1.index(str2, 10,32),"position in the statement\"")
```

```
str.center(40, 'a') :  ****this is string example....wow!!!****
str.count(sub, 4, 40) :  2
str.count(sub) :  1
"is is reapeated 2 times!"
-1
-1
"keyword exam is at  15 position in the statement"
```

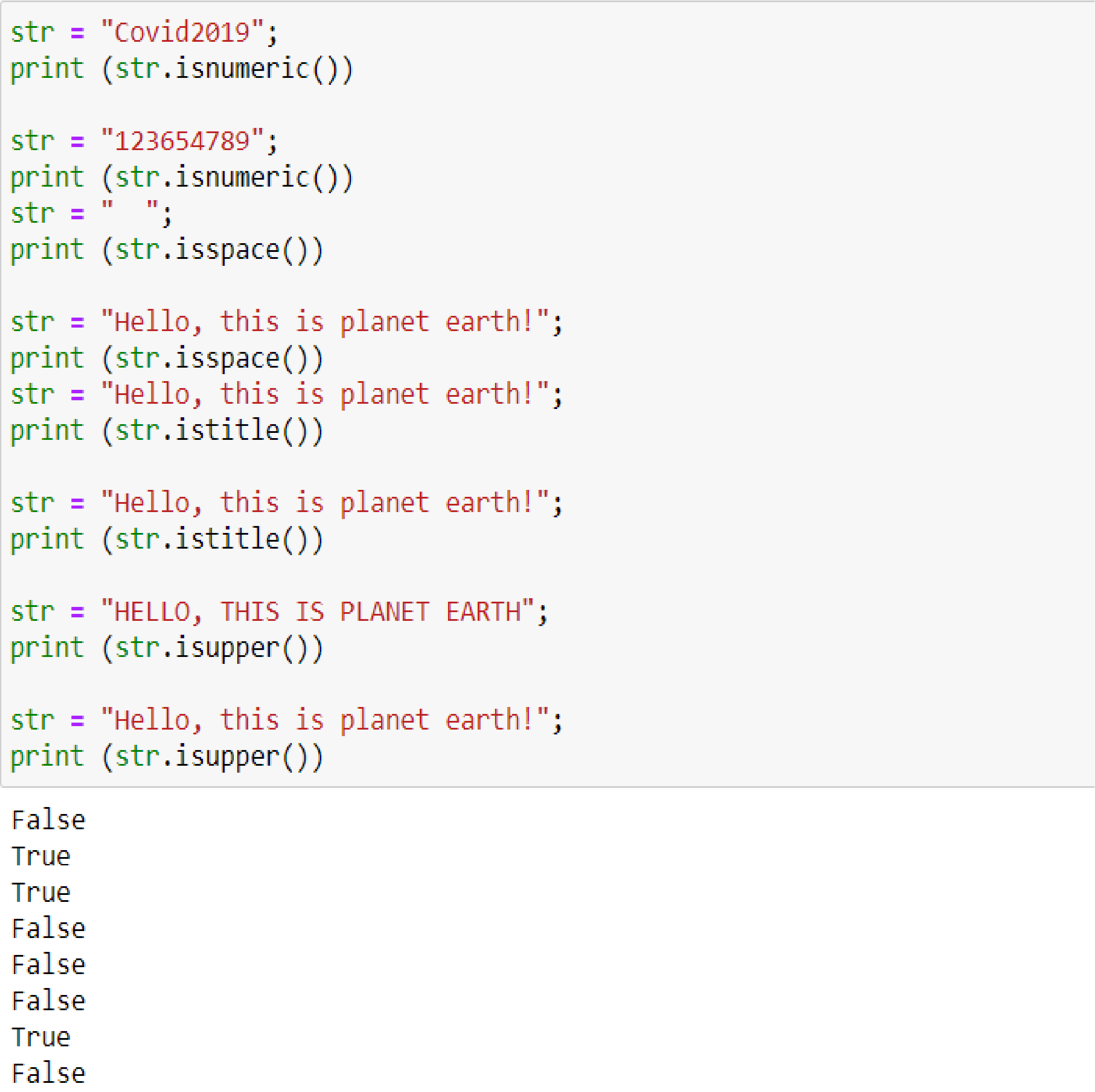
4. Program to show the function of isalnum()-checks whether the string consists of alphanumeric characters.



```
str = "this2021";  # No space in this string
print (str.isalnum())
str = "Hello, this is planet earth!";
print (str.isalnum())
str = "GulshanKumar";  # No space & digit in this string
print (str.isalpha())
str = "123456";  # Only digit in this string
print (str.isdigit())
str = "Hello, this is planet earth!";
print (str.isdigit())
str = "HELLO,THIS IS PLANET EARTH !";
print (str.islower())
str = "hello, this is planet earth!";
print (str.islower())
```

```
True
False
True
True
False
False
True
```

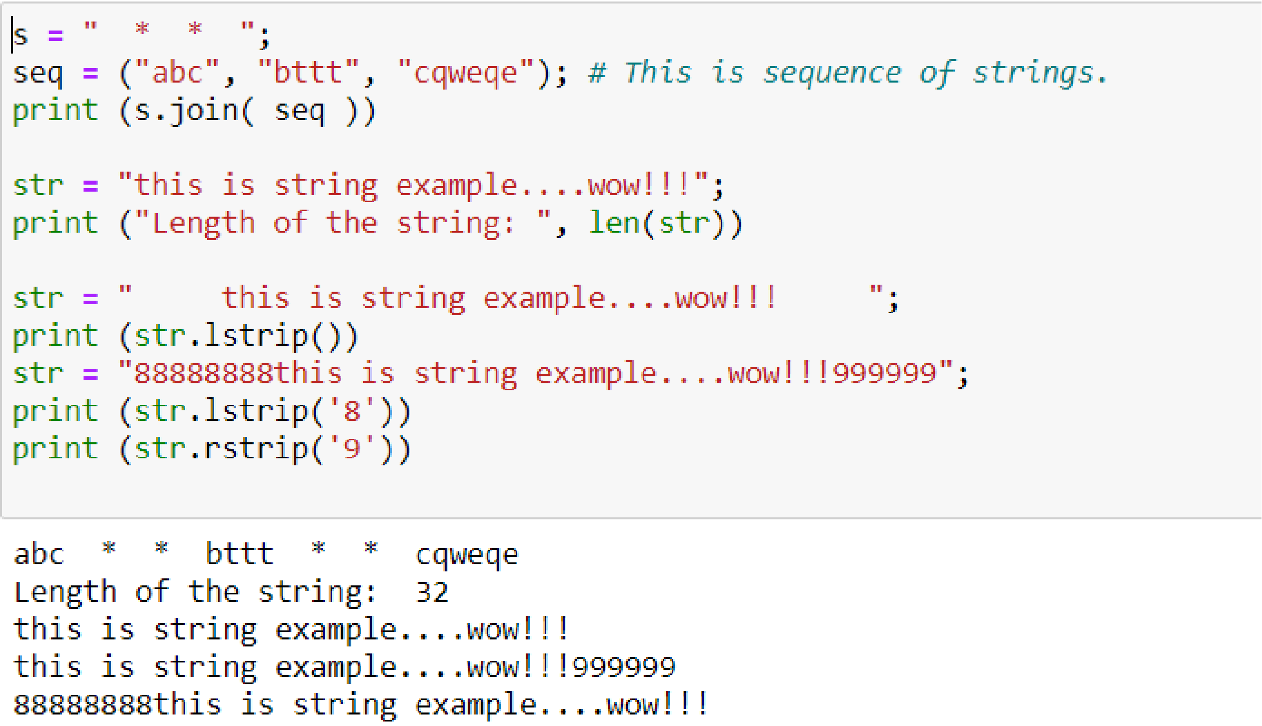
5.Program to show the function of isnumeric(), istitle(), isspace(), isupper()



```
str = "Covid2019";
print (str.isnumeric())

str = "123654789";
print (str.isnumeric())
str = "  ";
print (str.isspace())

str = "Hello, this is planet earth!";
print (str.isspace())
str = "Hello, this is planet earth!";
print (str.istitle())

str = "Hello, this is planet earth!";
print (str.istitle())

str = "HELLO, THIS IS PLANET EARTH";
print (str.isupper())

str = "Hello, this is planet earth!";
print (str.isupper())
```

```
False
True
True
False
False
False
True
False
```

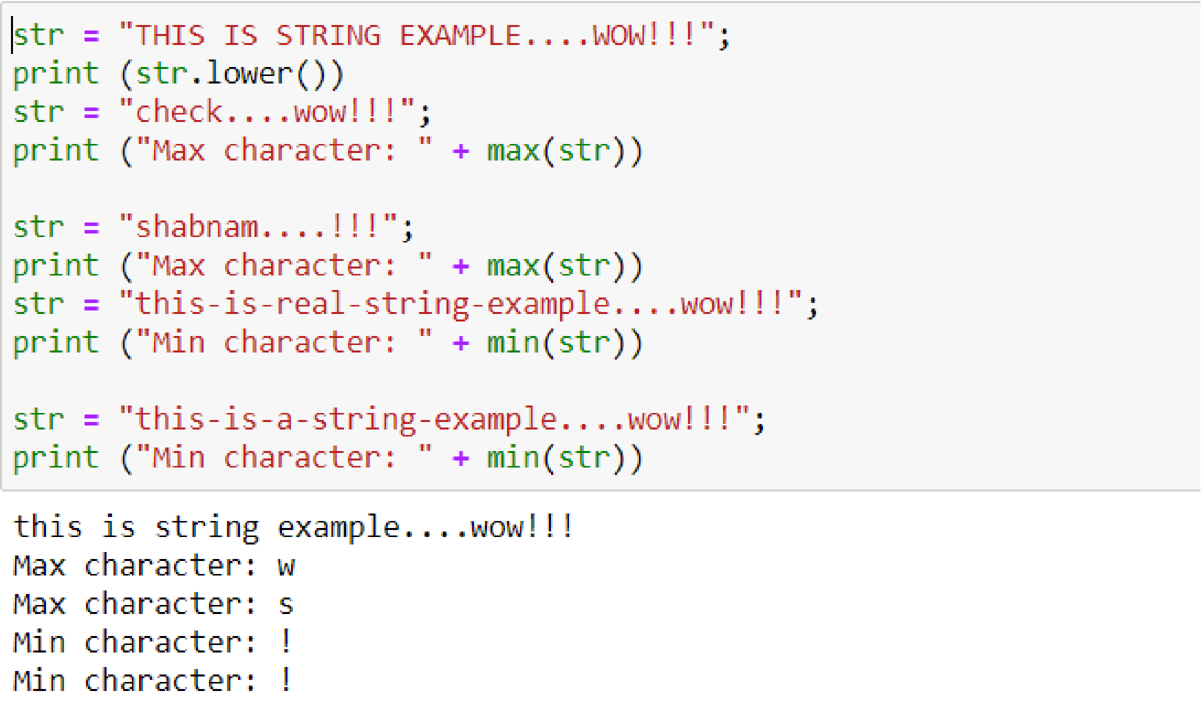
6.Program to show function join()-returns a string in which the string elements of sequence have been joined by str separator.



```
s = "  *  *  ";
seq = ("abc", "bttt", "cqweqe"); # This is sequence of strings.
print (s.join( seq ))

str = "this is string example....wow!!!";
print ("Length of the string: ", len(str))

str = "     this is string example....wow!!!     ";
print (str.lstrip())
str = "88888888this is string example....wow!!!999999";
print (str.lstrip('8'))
print (str.rstrip('9'))
```

```
abc  *  *  bttt  *  *  cqweqe
Length of the string:  32
this is string example....wow!!!
this is string example....wow!!!999999
88888888this is string example....wow!!!
```

7.Program to return largest character.



```
str = "THIS IS STRING EXAMPLE....WOW!!!";
print (str.lower())
str = "check....wow!!!";
print ("Max character: " + max(str))

str = "shabnam....!!!";
print ("Max character: " + max(str))
str = "this-is-real-string-example....wow!!!";
print ("Min character: " + min(str))

str = "this-is-a-string-example....wow!!!";
print ("Min character: " + min(str))
```

```
this is string example....wow!!!
Max character: w
Max character: s
Min character: !
Min character: !
```

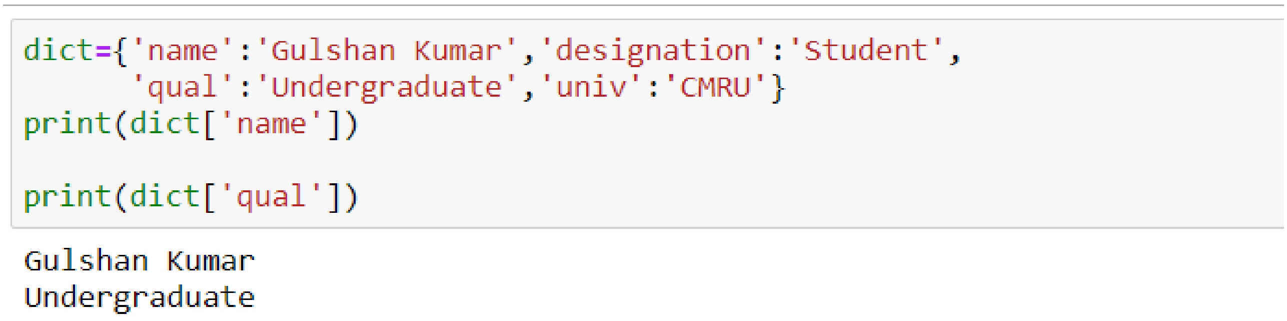
D. Dictionaries

Objective: To perform lists and dictionary operations.

Outcome: Tuple operations like Concatenation, Adding/Deleting elements, accessing elements, replacing elements ,Appending lists

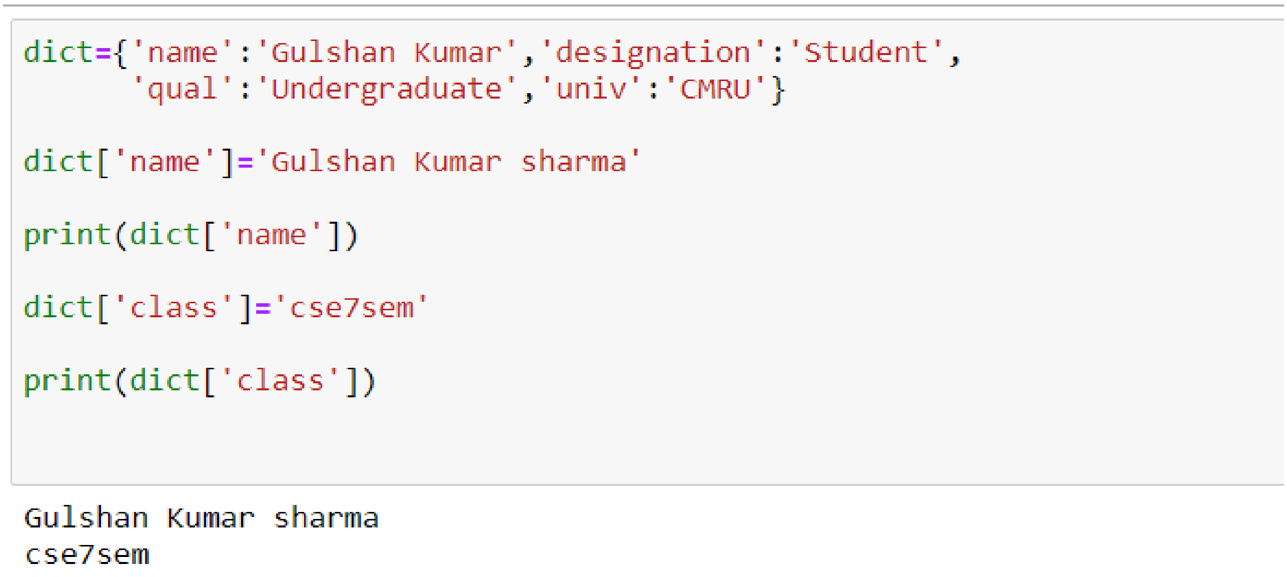
Conclusion: Successfully completed the Dictionaries and tuple operations

1.Program to implement Dictionary and access value using key.



```
dict={'name':'Gulshan Kumar','designation':'Student',
      'qual':'Undergraduate','univ':'CMRU'}
print(dict['name'])

print(dict['qual'])
```

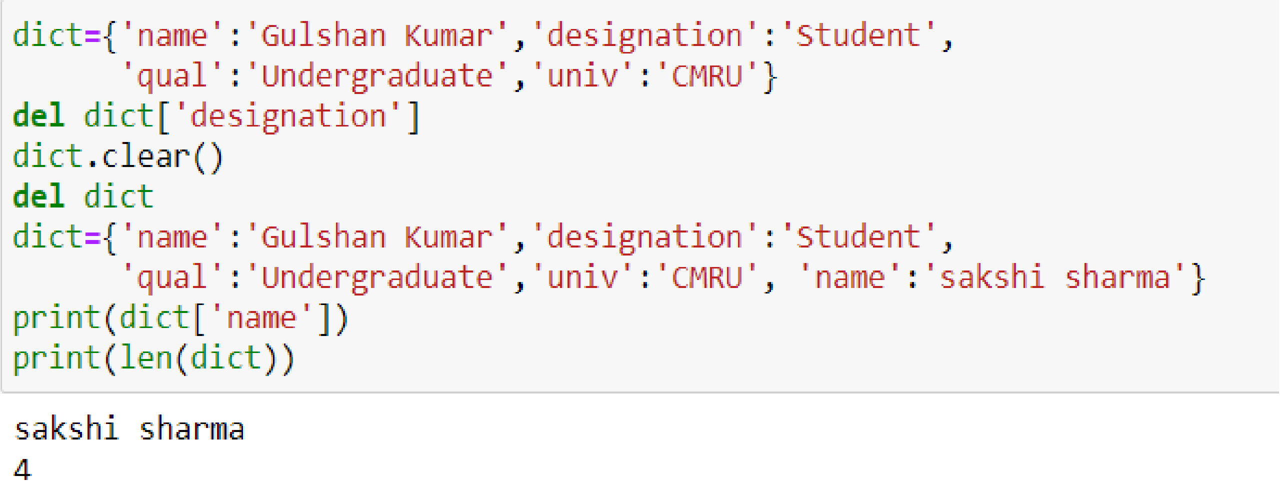
```
Gulshan Kumar
Undergraduate
```



```
dict={'name':'Gulshan Kumar','designation':'Student',
      'qual':'Undergraduate','univ':'CMRU'}

dict['name']='Gulshan Kumar sharma'

print(dict['name'])

dict['class']='cse7sem'

print(dict['class'])
```

```
Gulshan Kumar sharma
cse7sem
```

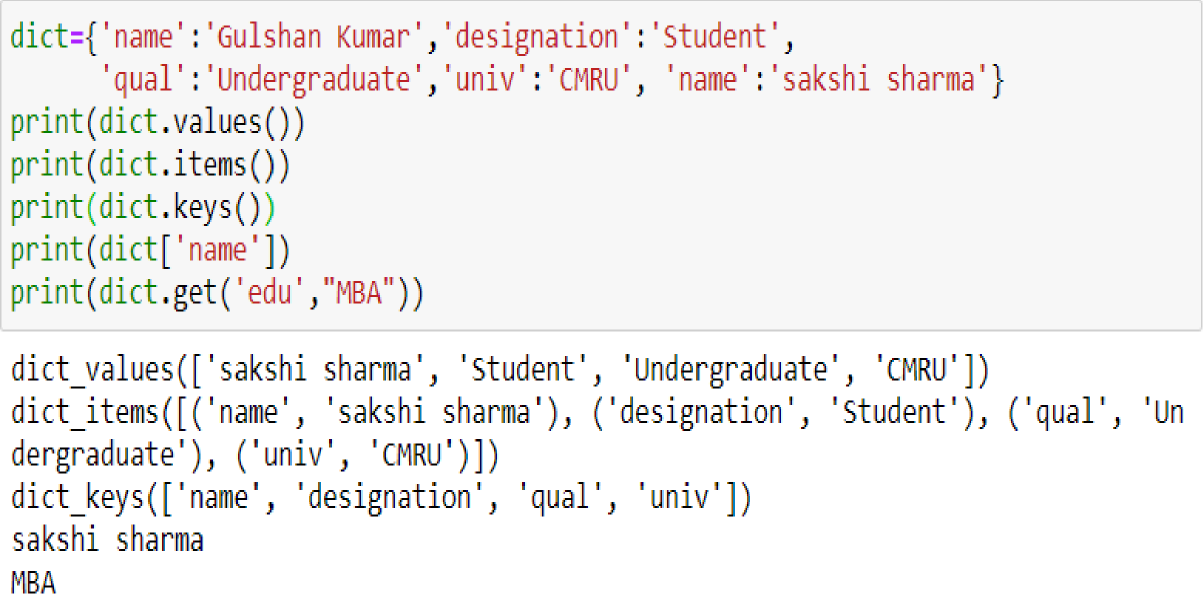
2.Program to show the function of del(), Len()



```
dict={'name':'Gulshan Kumar','designation':'Student',
      'qual':'Undergraduate','univ':'CMRU'}
del dict['designation']
dict.clear()
del dict
dict={'name':'Gulshan Kumar','designation':'Student',
      'qual':'Undergraduate','univ':'CMRU', 'name':'sakshi sharma'}
print(dict['name'])
print(len(dict))
```

```
sakshi sharma
4
```

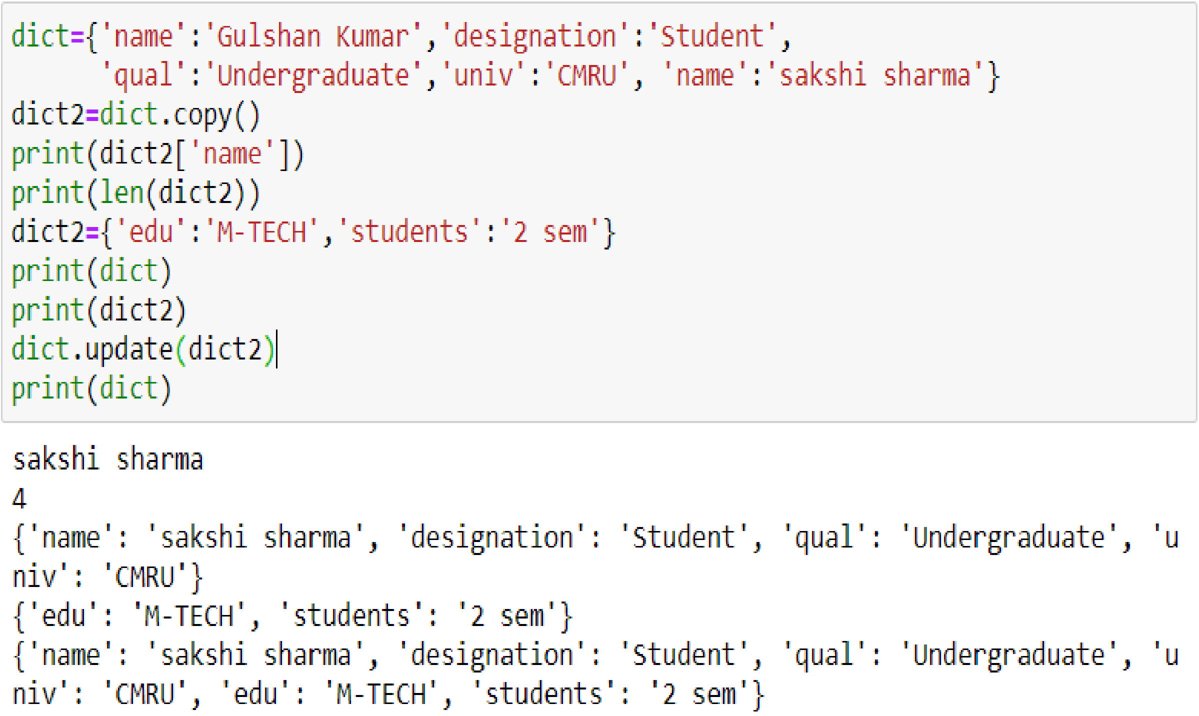
3. Program to access the values and items and ovverride the key.



```
dict={'name':'Gulshan Kumar','designation':'Student',
      'qual':'Undergraduate','univ':'CMRU', 'name':'sakshi sharma'}
print(dict.values())
print(dict.items())
print(dict.keys())
print(dict['name'])
print(dict.get('edu',"MBA"))
```

```
dict_values(['sakshi sharma', 'Student', 'Undergraduate', 'CMRU'])
dict_items([('name', 'sakshi sharma'), ('designation', 'Student'), ('qual', 'Un
dergraduate'), ('univ', 'CMRU')])
dict_keys(['name', 'designation', 'qual', 'univ'])
sakshi sharma
MBA
```

4.Program to copy one dictionary item to another dictionary.



```
dict={'name':'Gulshan Kumar','designation':'Student',
      'qual':'Undergraduate','univ':'CMRU', 'name':'sakshi sharma'}
dict2=dict.copy()
print(dict2['name'])
print(len(dict2))
dict2={'edu':'M-TECH','students':'2 sem'}
print(dict)
print(dict2)
dict.update(dict2)
print(dict)
```

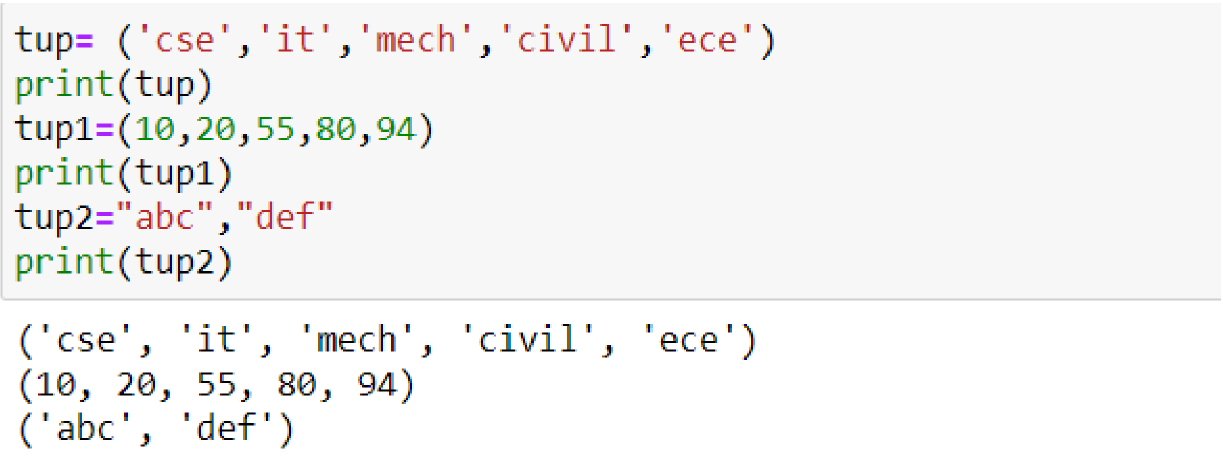
```
sakshi sharma
4
{'name': 'sakshi sharma', 'designation': 'Student', 'qual': 'Undergraduate', 'u
niv': 'CMRU'}
{'edu': 'M-TECH', 'students': '2 sem'}
{'name': 'sakshi sharma', 'designation': 'Student', 'qual': 'Undergraduate', 'u
niv': 'CMRU', 'edu': 'M-TECH', 'students': '2 sem'}
```

E. Tuples

Objective: To perform lists and dictionary operations.

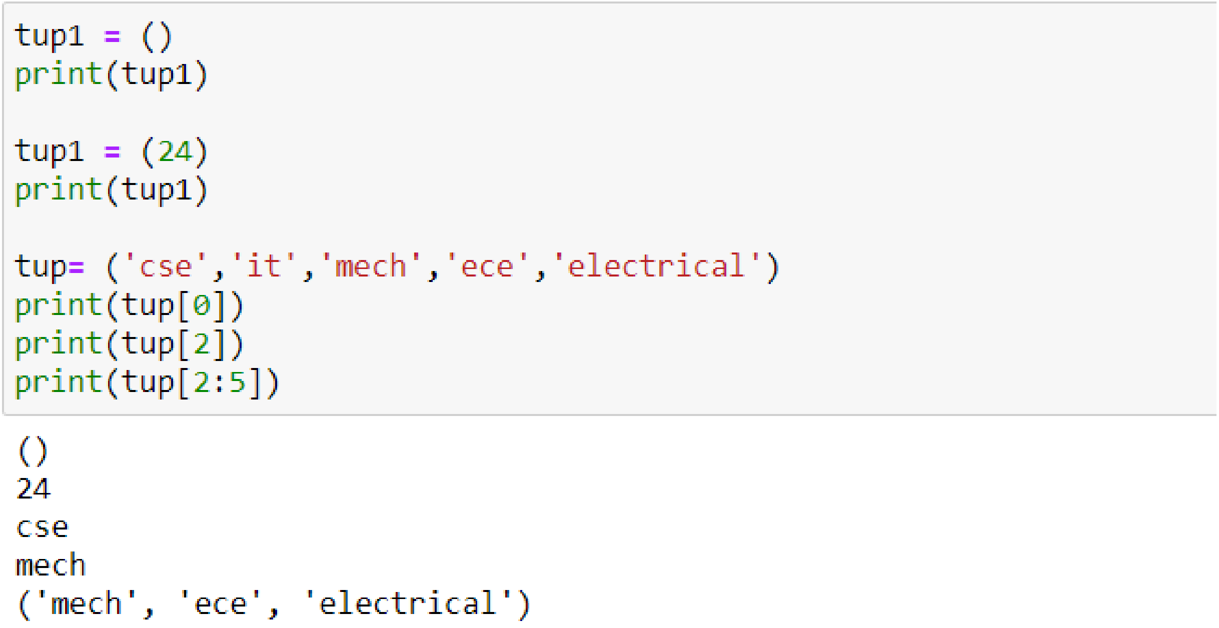
Outcome: Tuple operations like Concatenation, Adding/Deleting elements, accessing elements, replacing elements ,Appending lists Conclusion: Successfully completed the Dictionaries and tuple operations

1.Program to create a tuple and access the elements of tuple.



```
tup= ('cse','it','mech','civil','ece')
print(tup)
tup1=(10,20,55,80,94)
print(tup1)
tup2="abc","def"
print(tup2)
```

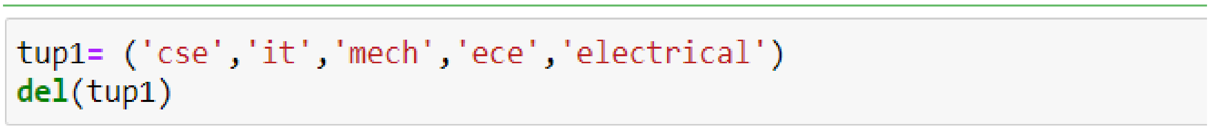
```
('cse', 'it', 'mech', 'civil', 'ece')
(10, 20, 55, 80, 94)
('abc', 'def')
```



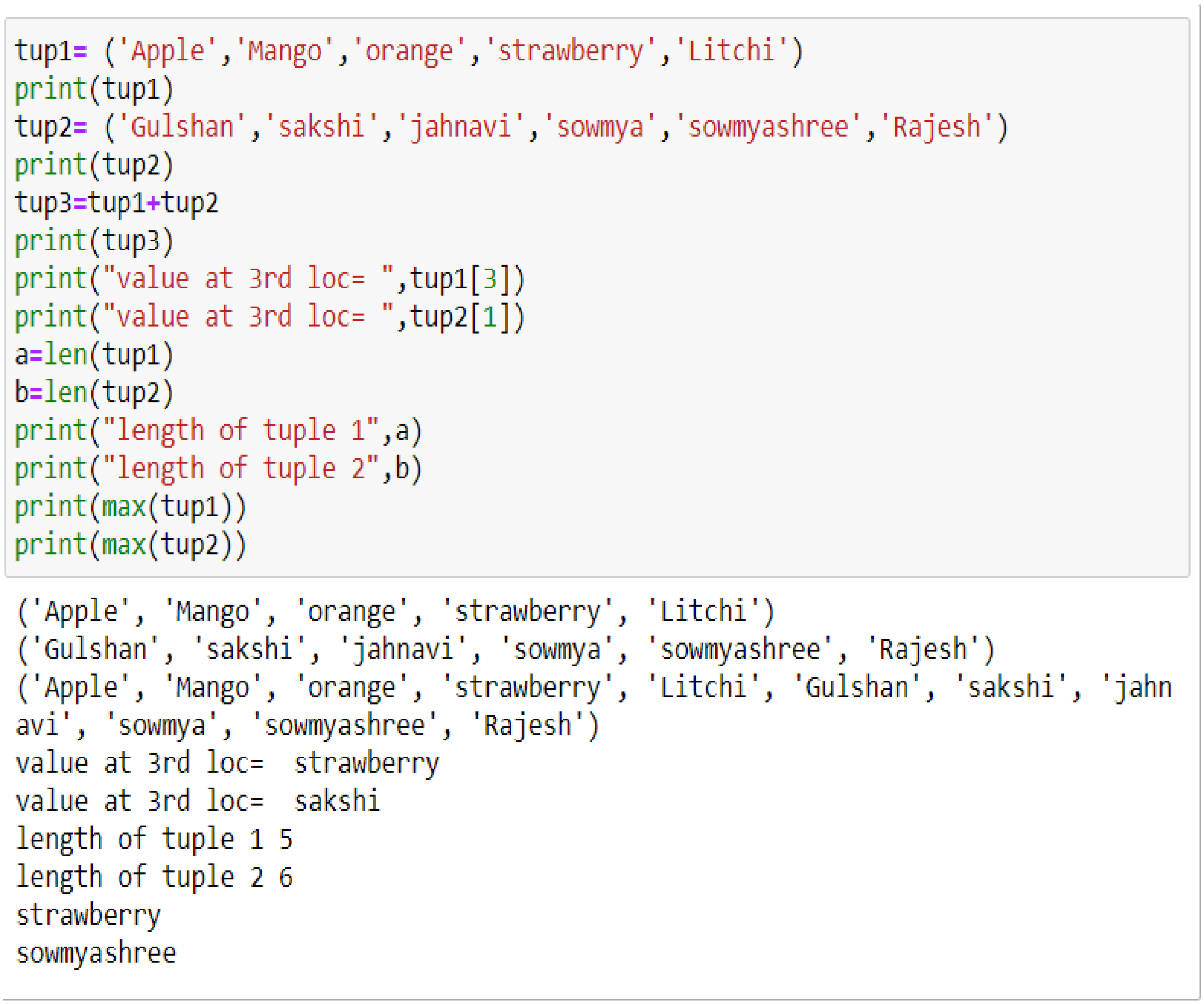
```
tup1 = ()
print(tup1)

tup1 = (24)
print(tup1)

tup= ('cse','it','mech','ece','electrical')
print(tup[0])
print(tup[2])
print(tup[2:5])
```

```
()
24
cse
mech
('mech', 'ece', 'electrical')
```

2.Program to del a tuple.



```
tup1= ('cse','it','mech','ece','electrical')
del(tup1)
```

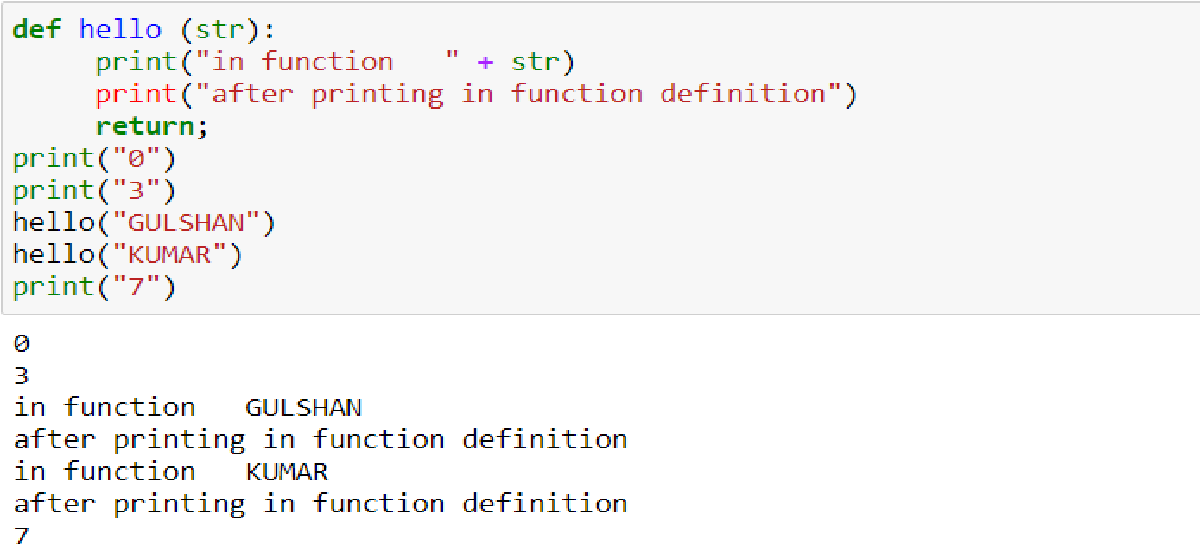


```
tup1= ('Apple','Mango','orange','strawberry','Litchi')
print(tup1)
tup2= ('Gulshan','sakshi','jahnavi','sowmya','sowmyashree','Rajesh')
print(tup2)
tup3=tup1+tup2
print(tup3)
print("value at 3rd loc= ",tup1[3])
print("value at 3rd loc= ",tup2[1])
a=len(tup1)
b=len(tup2)
print("length of tuple 1",a)
print("length of tuple 2",b)
print(max(tup1))
print(max(tup2))
```

```
('Apple', 'Mango', 'orange', 'strawberry', 'Litchi')
('Gulshan', 'sakshi', 'jahnavi', 'sowmya', 'sowmyashree', 'Rajesh')
('Apple', 'Mango', 'orange', 'strawberry', 'Litchi', 'Gulshan', 'sakshi', 'jahn
avi', 'sowmya', 'sowmyashree', 'Rajesh')
value at 3rd loc=  strawberry
value at 3rd loc=  sakshi
length of tuple 1 5
length of tuple 2 6
strawberry
sowmyashree
```

F. Function

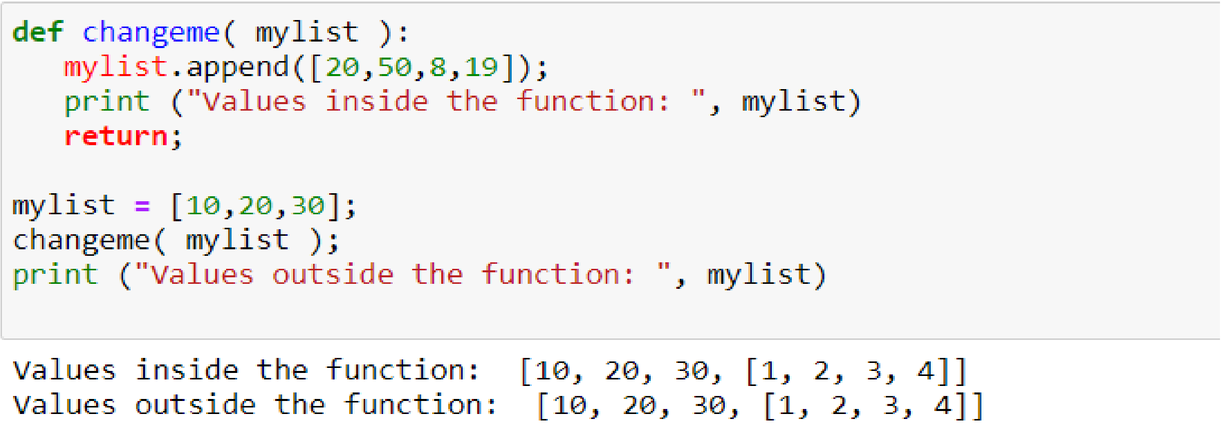
1.Program to understand the syntax of function.



```
def hello (str):
     print("in function   " + str)
     print("after printing in function definition")
     return;
print("0")
print("3")
hello("GULSHAN")
hello("KUMAR")
print("7")
```

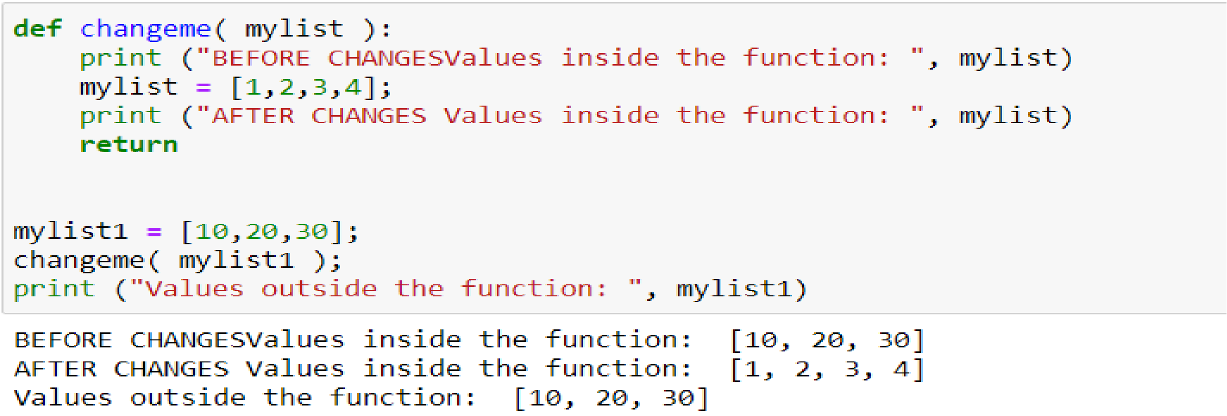
```
0
3
in function   GULSHAN
after printing in function definition
in function   KUMAR
after printing in function definition
7
```

2.Program to print a list using function.



```
def changeme( mylist ):
   mylist.append([20,50,8,19]);
   print ("Values inside the function: ", mylist)
   return;

mylist = [10,20,30];
changeme( mylist );
print ("Values outside the function: ", mylist)
```

```
Values inside the function:  [10, 20, 30, [1, 2, 3, 4]]
Values outside the function:  [10, 20, 30, [1, 2, 3, 4]]
```



```
def changeme( mylist ):
    print ("BEFORE CHANGESValues inside the function: ", mylist)
    mylist = [1,2,3,4];
    print ("AFTER CHANGES Values inside the function: ", mylist)
    return


mylist1 = [10,20,30];
changeme( mylist1 );
print ("Values outside the function: ", mylist1)
```

```
BEFORE CHANGESValues inside the function:  [10, 20, 30]
AFTER CHANGES Values inside the function:  [1, 2, 3, 4]
Values outside the function:  [10, 20, 30]
```

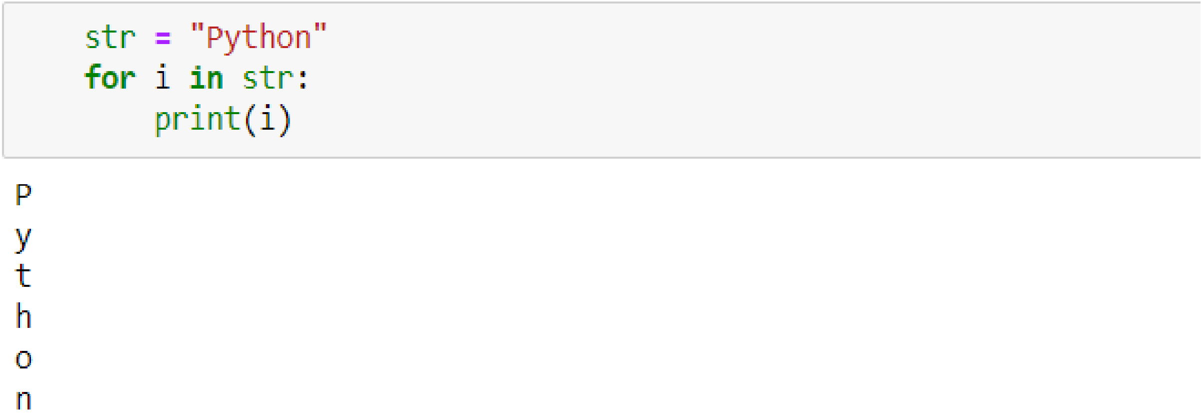
G .Loops

Objective: To perform loops operations using break, continue

Outcome: Working with loops using continue break

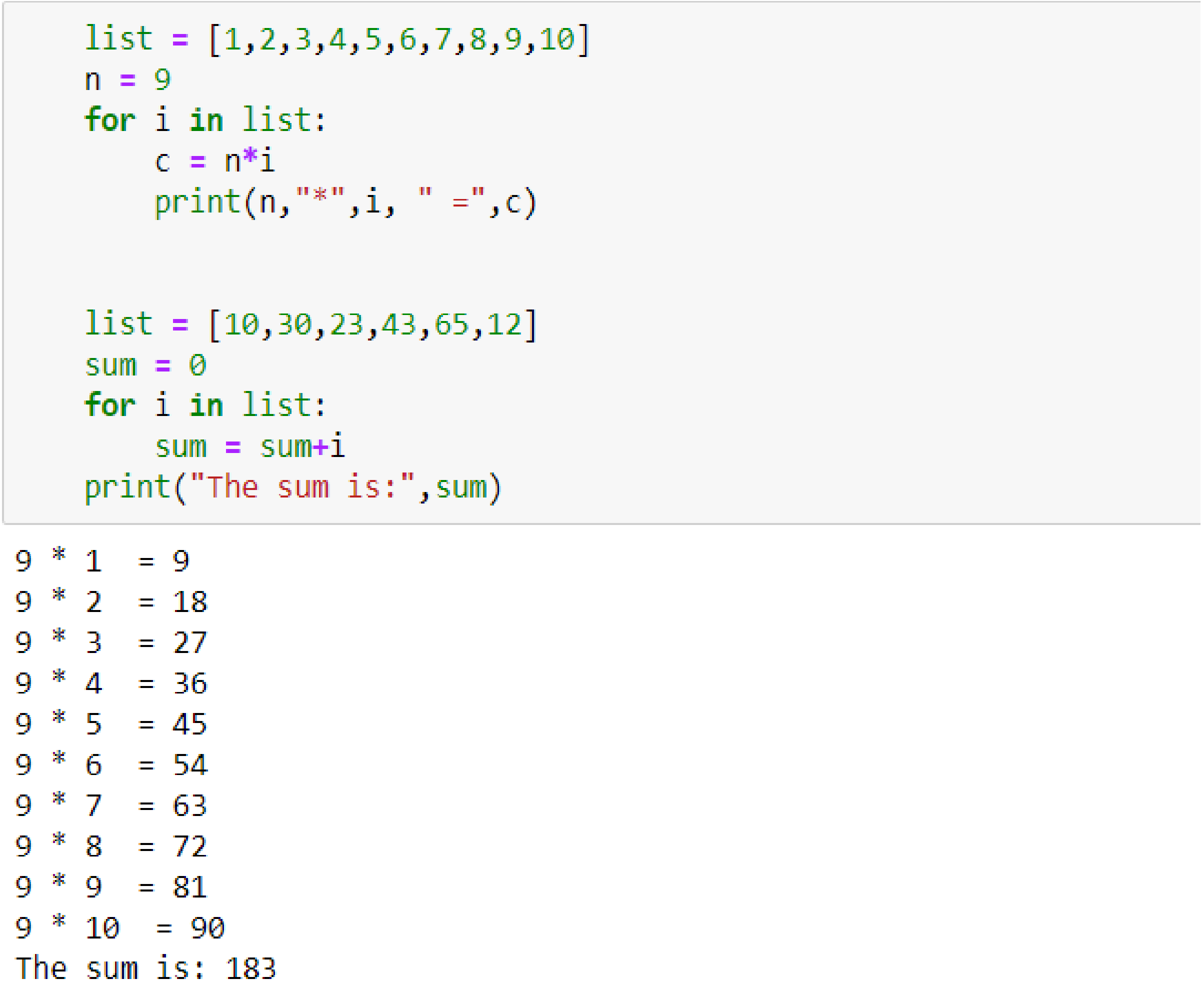
Conclusion: Successfully implemented file operations

1.Program to Calculate multiplication table.



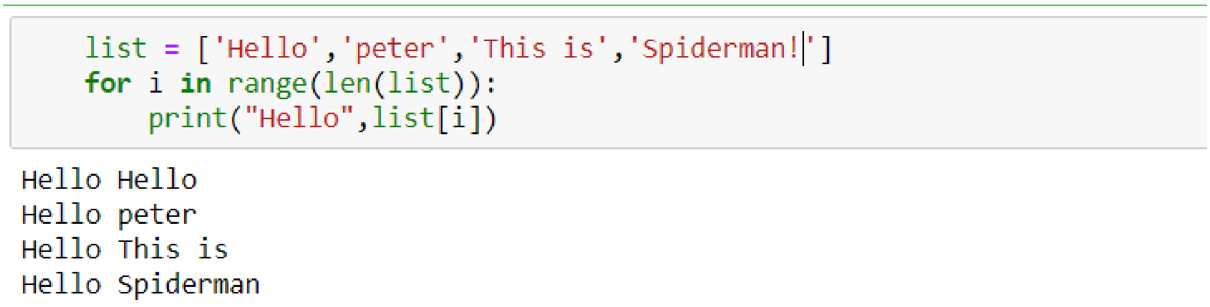
```
str = "Python"
for i in str:
    print(i)
```

```
P
y
t
h
o
n
```



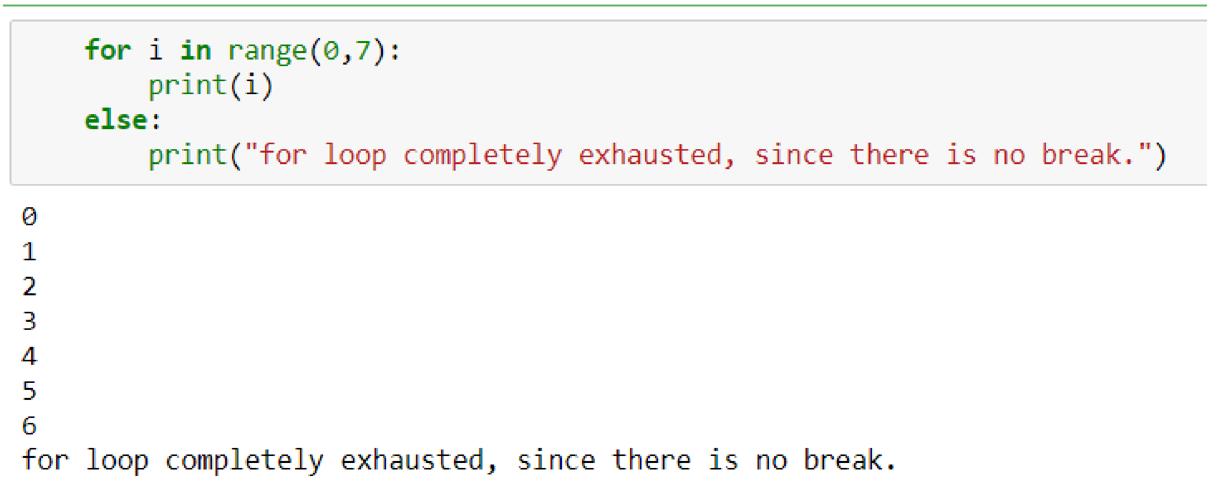
```
list = [1,2,3,4,5,6,7,8,9,10]
n = 9
for i in list:
    c = n*i
    print(n,"*",i, " =",c)


list = [10,30,23,43,65,12]
sum = 0
for i in list:
    sum = sum+i
print("The sum is:",sum)
```

```
9 * 1  = 9
9 * 2  = 18
9 * 3  = 27
9 * 4  = 36
9 * 5  = 45
9 * 6  = 54
9 * 7  = 63
9 * 8  = 72
9 * 9  = 81
9 * 10  = 90
The sum is: 183
```



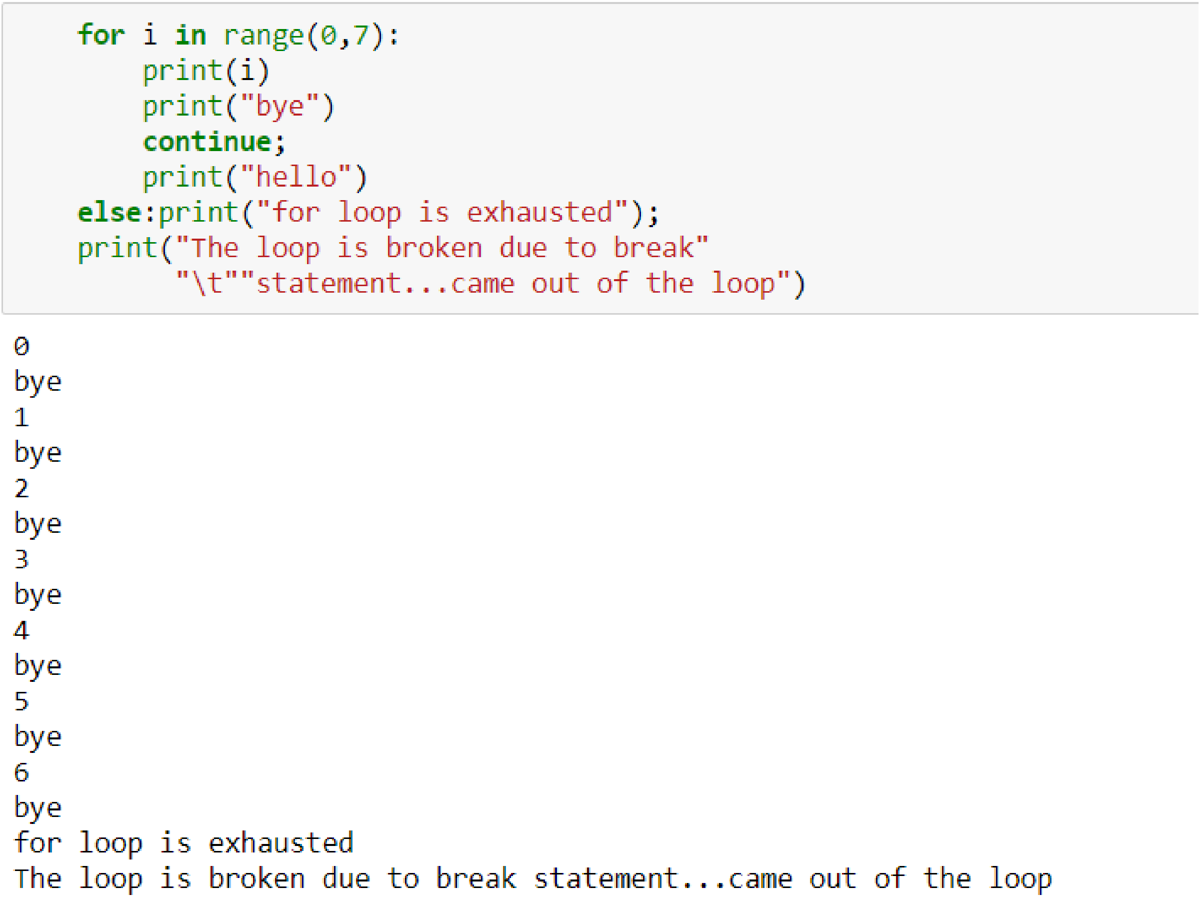
```
list = ['Hello','peter','This is','Spiderman!']
for i in range(len(list)):
    print("Hello",list[i])
```

```
Hello Hello
Hello peter
Hello This is
Hello Spiderman
```



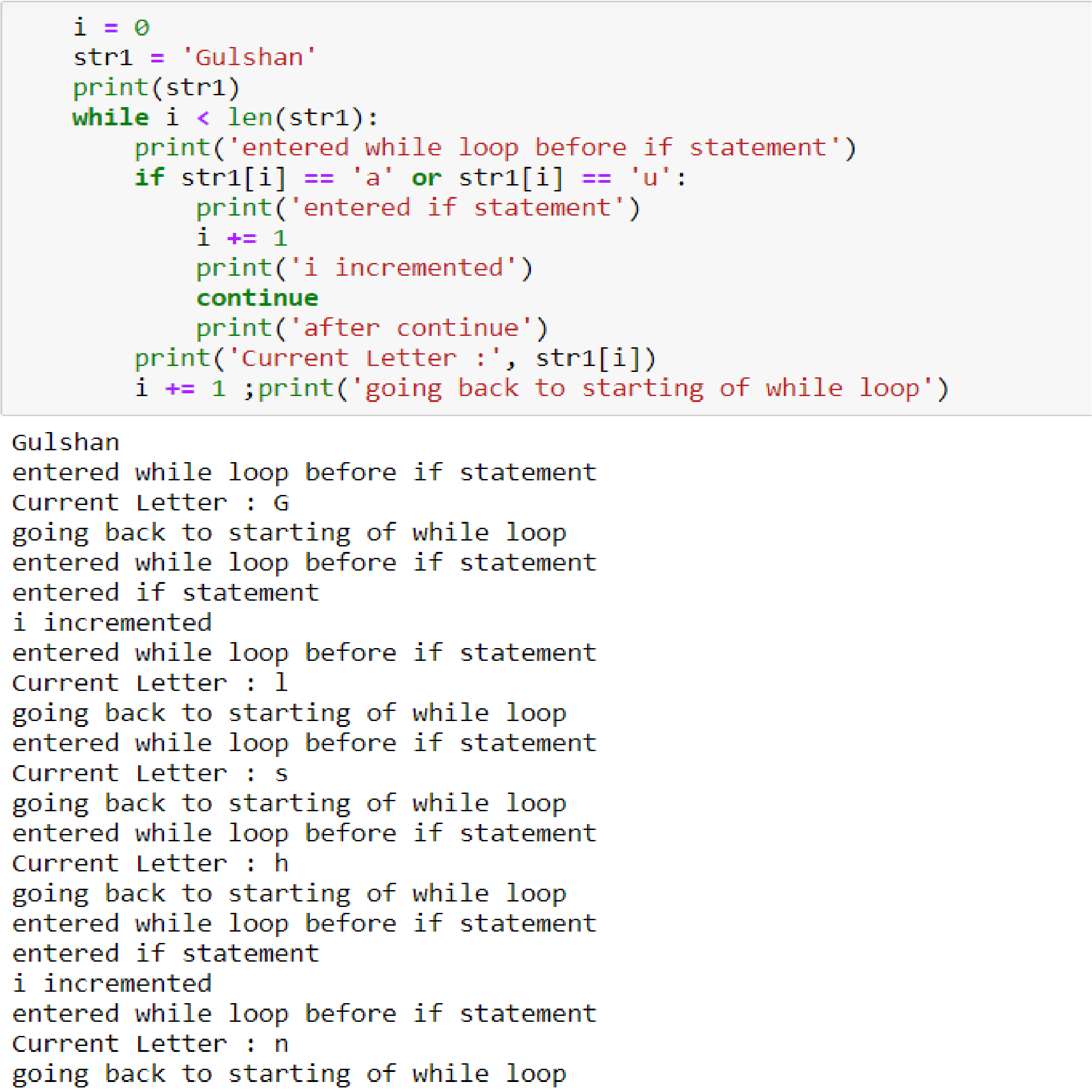
```
for i in range(0,7):
    print(i)
else:
    print("for loop completely exhausted, since there is no break.")
```

```
0
1
2
3
4
5
6
for loop completely exhausted, since there is no break.
```



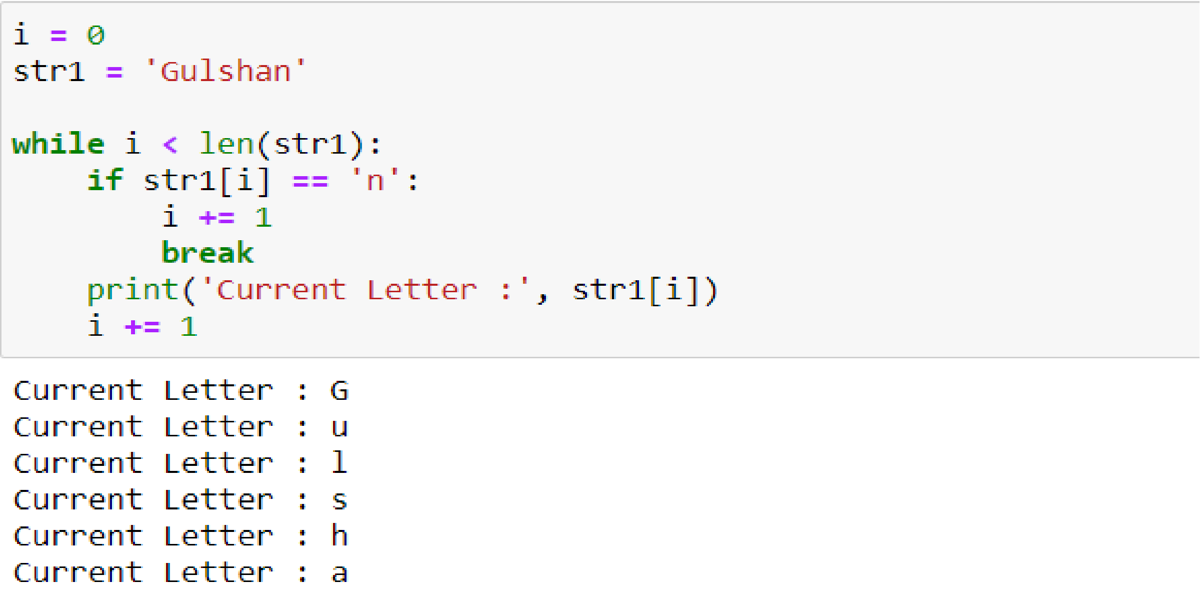
```
for i in range(0,7):
    print(i)
    print("bye")
    continue;
    print("hello")
else:print("for loop is exhausted");
print("The loop is broken due to break"
      "\t""statement...came out of the loop")
```

```
0
bye
1
bye
2
bye
3
bye
4
bye
5
bye
6
bye
for loop is exhausted
The loop is broken due to break statement...came out of the loop
```



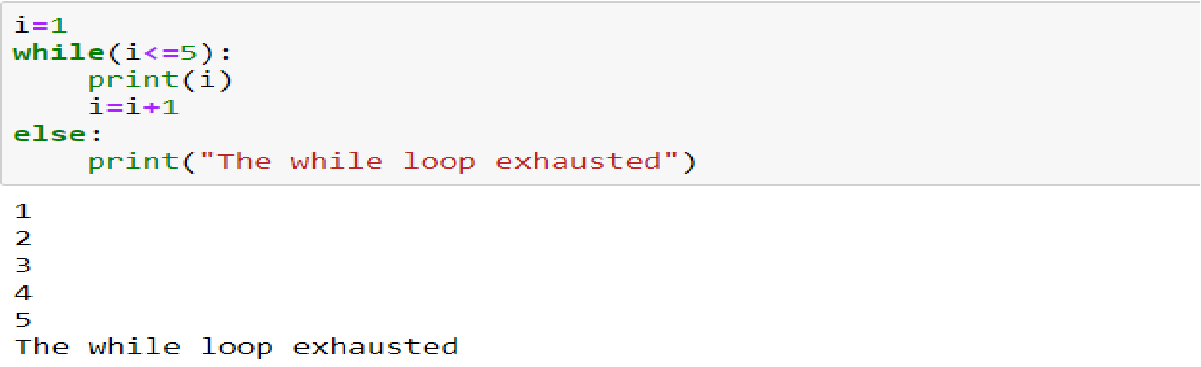
```
    i = 0
    str1 = 'Gulshan'
    print(str1)
    while i < len(str1):
        print('entered while loop before if statement')
        if str1[i] == 'a' or str1[i] == 'u':
            print('entered if statement')
            i += 1
            print('i incremented')
            continue
            print('after continue')
        print('Current Letter :', str1[i])
        i += 1 ;print('going back to starting of while loop')
```

```
Gulshan
entered while loop before if statement
Current Letter : G
going back to starting of while loop
entered while loop before if statement
entered if statement
i incremented
entered while loop before if statement
Current Letter : l
going back to starting of while loop
entered while loop before if statement
Current Letter : s
going back to starting of while loop
entered while loop before if statement
Current Letter : h
going back to starting of while loop
entered while loop before if statement
entered if statement
i incremented
entered while loop before if statement
Current Letter : n
going back to starting of while loop
```



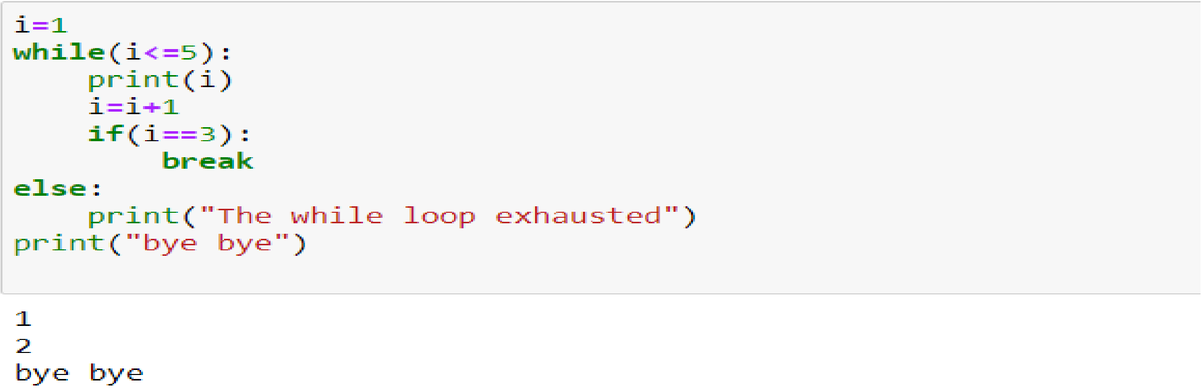
```
i = 0
str1 = 'Gulshan'

while i < len(str1):
    if str1[i] == 'n':
        i += 1
        break
    print('Current Letter :', str1[i])
    i += 1
```

```
Current Letter : G
Current Letter : u
Current Letter : l
Current Letter : s
Current Letter : h
Current Letter : a
```



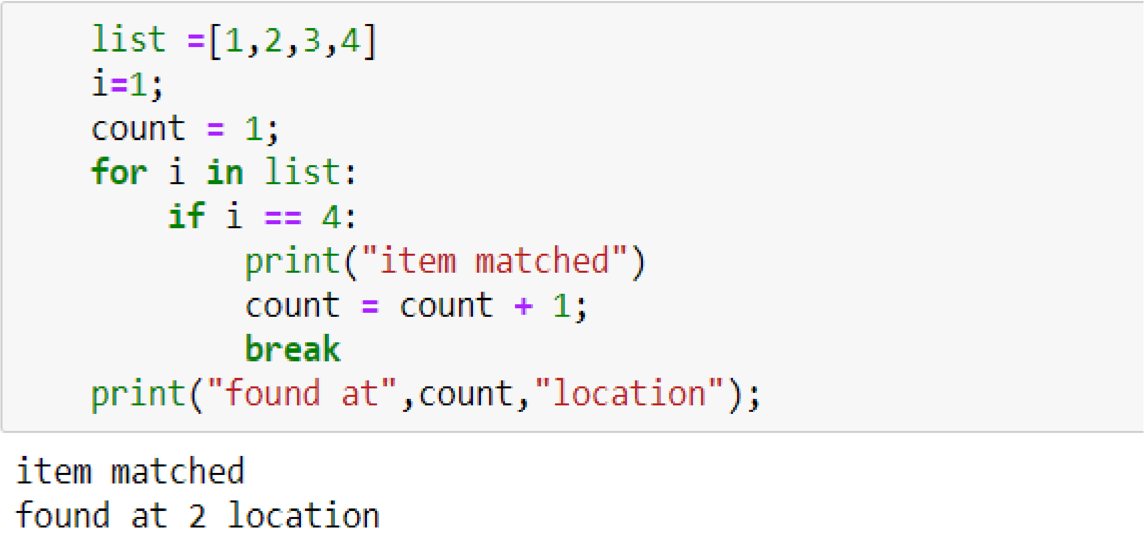
```
i=1
while(i<=5):
    print(i)
    i=i+1
else:
    print("The while loop exhausted")
```

```
1
2
3
4
5
The while loop exhausted
```



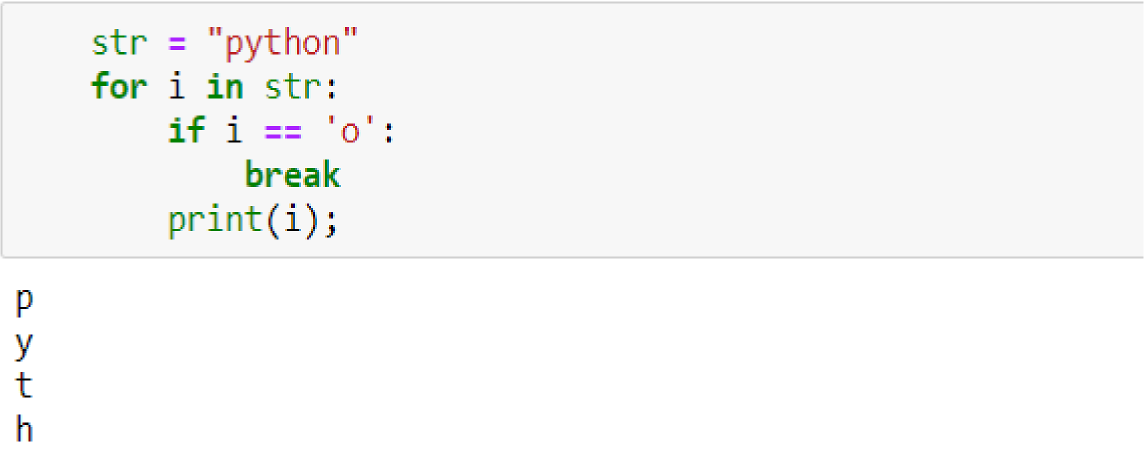
```
i=1
while(i<=5):
    print(i)
    i=i+1
    if(i==3):
        break
else:
    print("The while loop exhausted")
print("bye bye")
```

```
1
2
bye bye
```



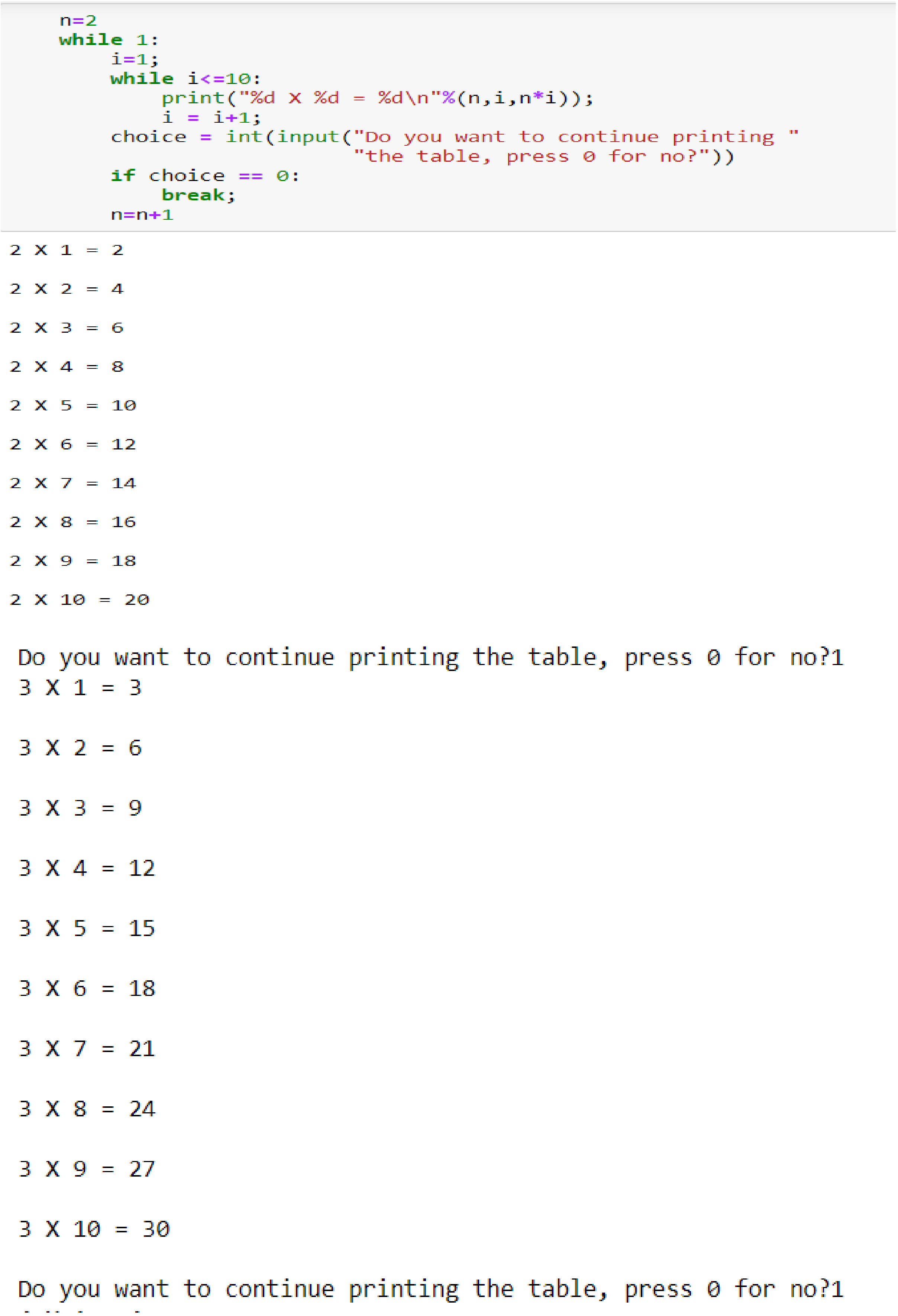
```
    list =[1,2,3,4]
    i=1;
    count = 1;
    for i in list:
        if i == 4:
            print("item matched")
            count = count + 1;
            break
    print("found at",count,"location");
```

```
item matched
found at 2 location
```



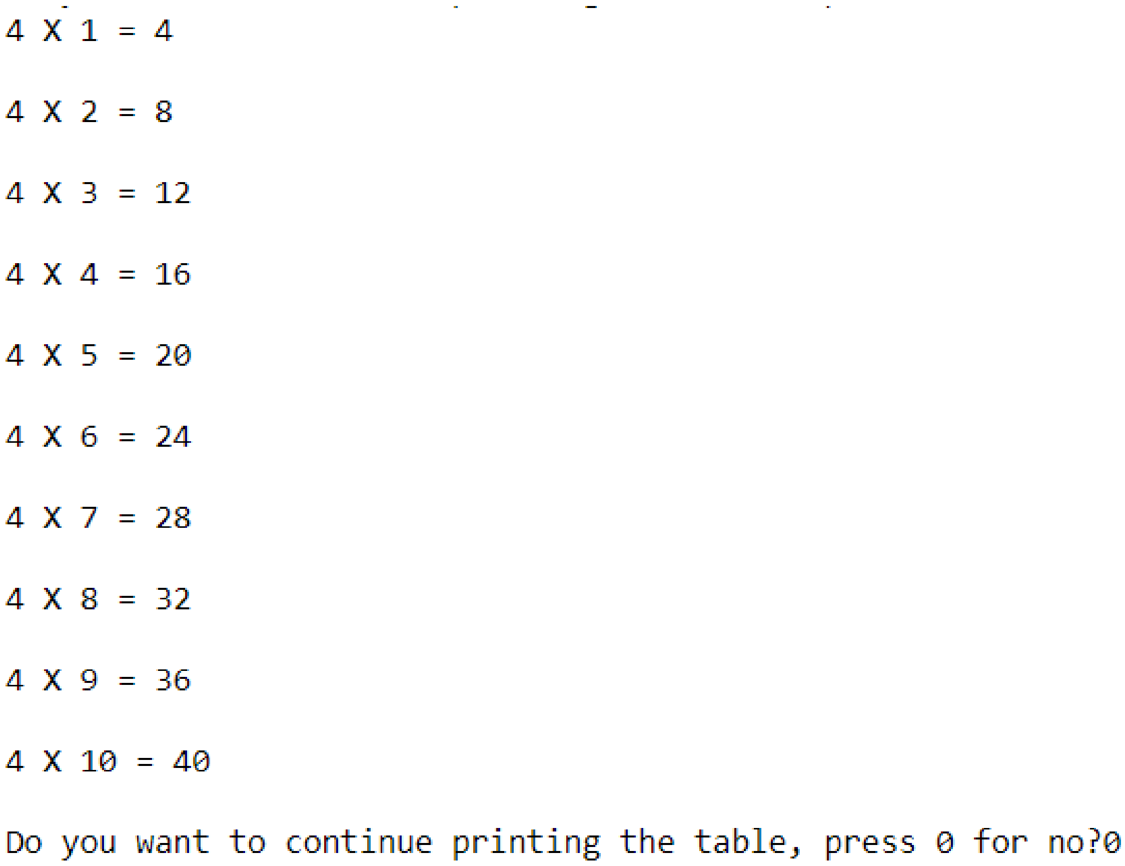
```
    str = "python"
    for i in str:
        if i == 'o':
            break
        print(i);
```

```
p
y
t
h
```

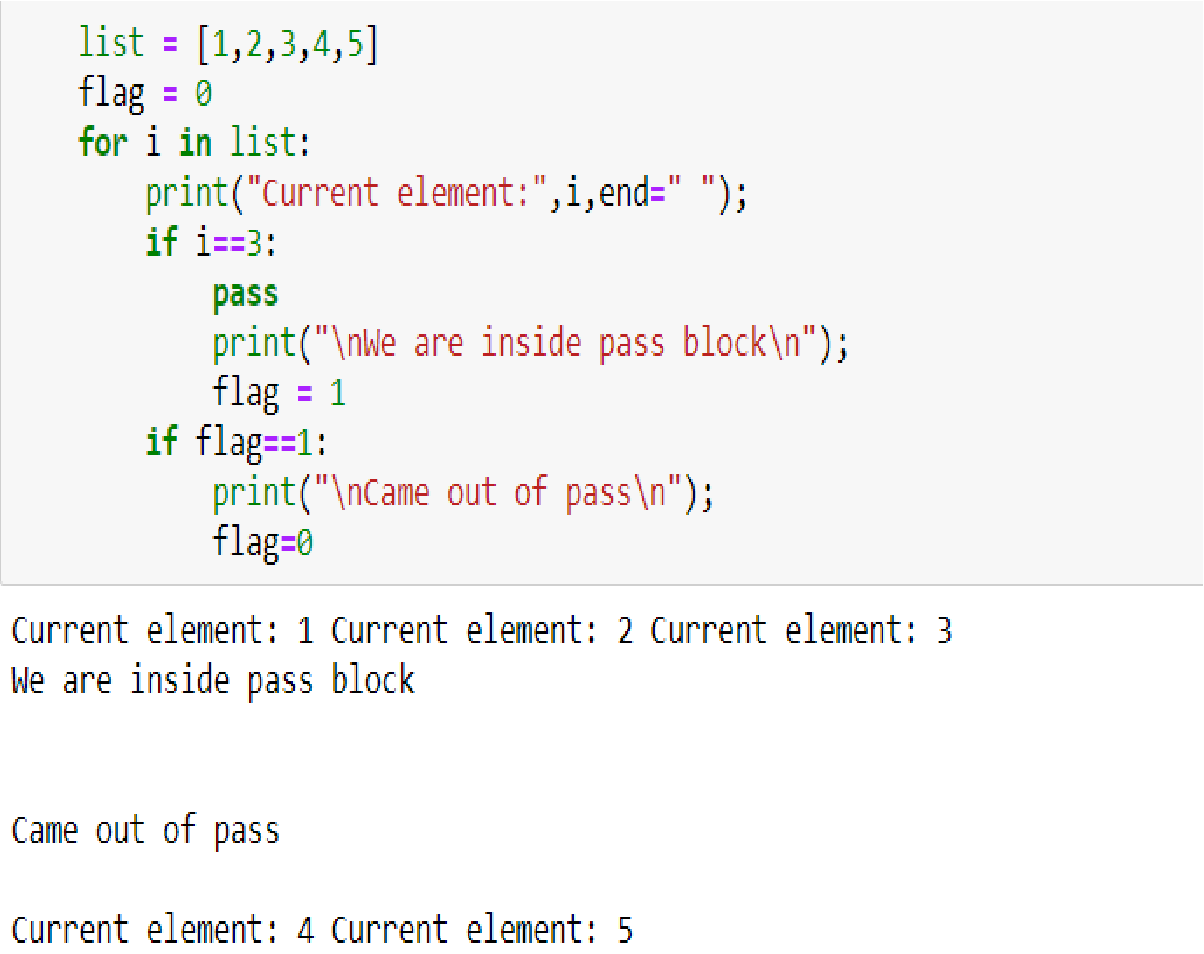


```python
n=2
while 1:
    i=1;
    while i<=10:
        print("%d X %d = %d\n"%(n,i,n*i));
        i = i+1;
    choice = int(input("Do you want to continue printing "
                       "the table, press 0 for no?"))
    if choice == 0:
        break;
    n=n+1
```

```
2 X 1 = 2

2 X 2 = 4

2 X 3 = 6

2 X 4 = 8

2 X 5 = 10

2 X 6 = 12

2 X 7 = 14

2 X 8 = 16

2 X 9 = 18

2 X 10 = 20

Do you want to continue printing the table, press 0 for no?1
3 X 1 = 3

3 X 2 = 6

3 X 3 = 9

3 X 4 = 12

3 X 5 = 15

3 X 6 = 18

3 X 7 = 21

3 X 8 = 24

3 X 9 = 27

3 X 10 = 30

Do you want to continue printing the table, press 0 for no?1
```

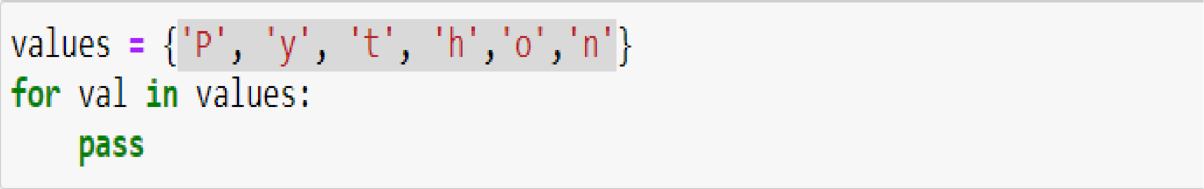


```
4 X 1 = 4

4 X 2 = 8

4 X 3 = 12

4 X 4 = 16

4 X 5 = 20

4 X 6 = 24

4 X 7 = 28

4 X 8 = 32

4 X 9 = 36

4 X 10 = 40

Do you want to continue printing the table, press 0 for no?0
```

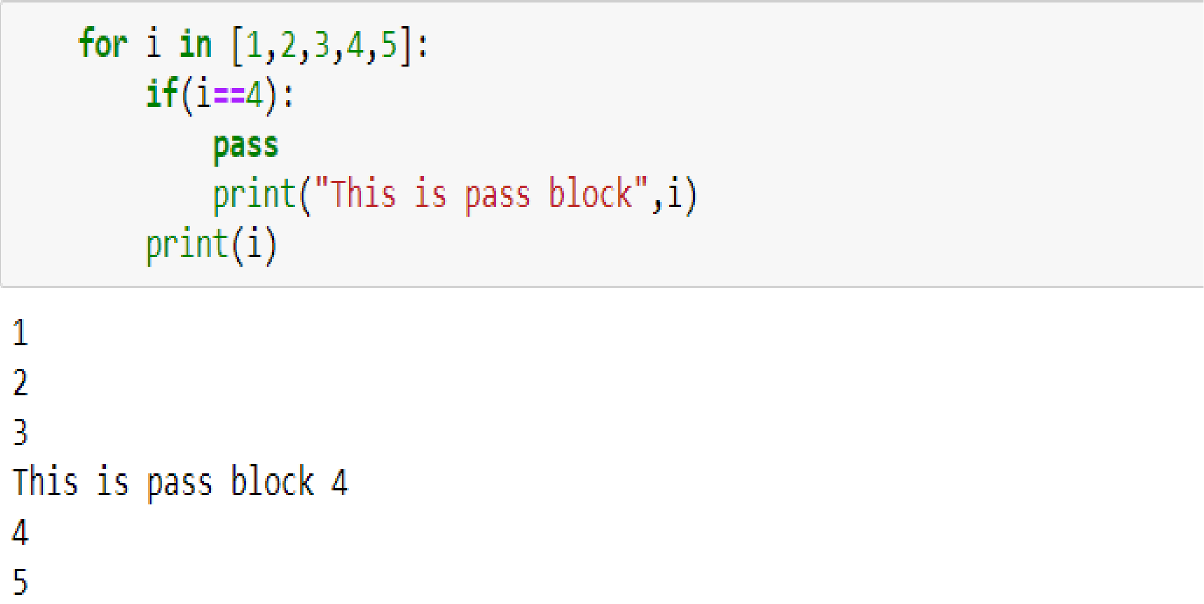


```
    list = [1,2,3,4,5]
    flag = 0
    for i in list:
        print("Current element:",i,end=" ");
        if i==3:
            pass
            print("\nWe are inside pass block\n");
            flag = 1
        if flag==1:
            print("\nCame out of pass\n");
            flag=0
```

```
Current element: 1 Current element: 2 Current element: 3
We are inside pass block


Came out of pass

Current element: 4 Current element: 5
```



```
values = {'P', 'y', 't', 'h','o','n'}
for val in values:
    pass
```



```
    for i in [1,2,3,4,5]:
        if(i==4):
            pass
            print("This is pass block",i)
        print(i)
```

```
1
2
3
This is pass block 4
4
5
```

H .Numpy

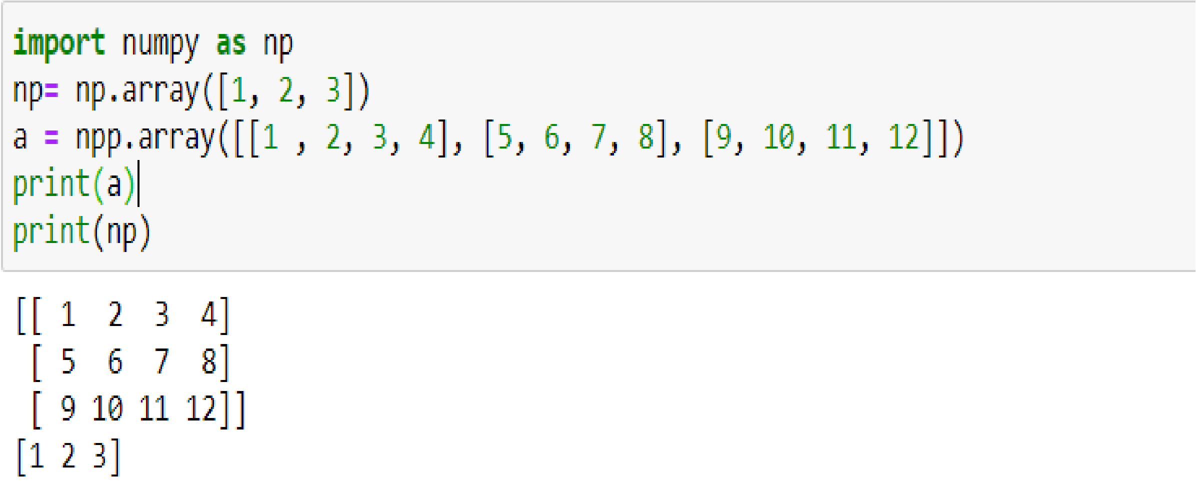
Objective: To perform Numpy operations

Outcome: File operations like Addition, Power, Reciprocal, Multiplication, Modulus, Division

Conclusion: Successfully implemented Numpy operations

1. Introduction to Numpy

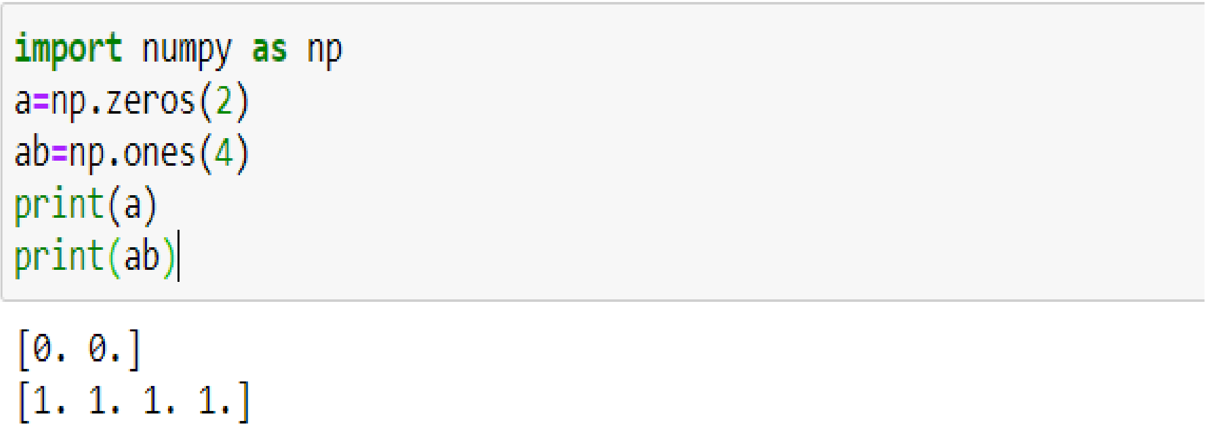
Program 1: Import Numpy and implement array.



```
import numpy as np
np= np.array([1, 2, 3])
a = npp.array([[1 , 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]])
print(a)
print(np)
```

```
[[ 1  2  3  4]
 [ 5  6  7  8]
 [ 9 10 11 12]]
[1 2 3]
```

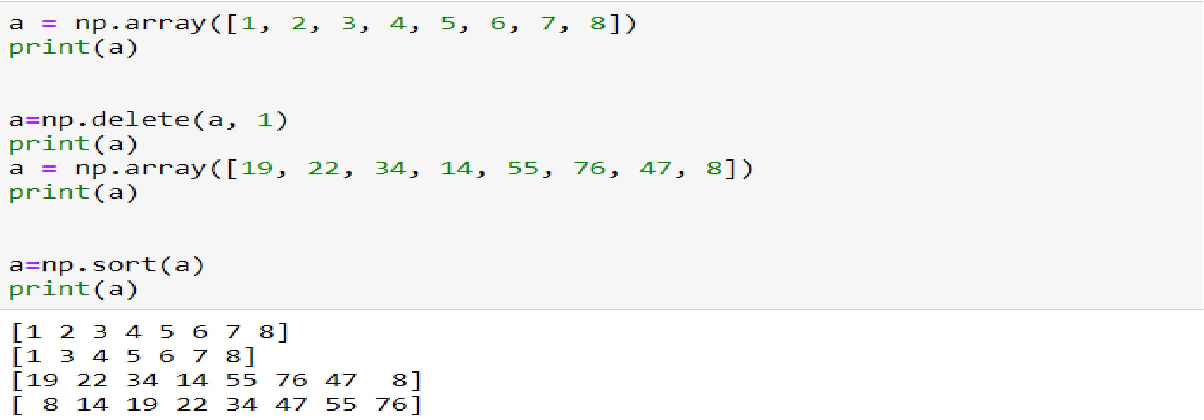
Program 2: Create an array with only zeros and ones.



```
import numpy as np
a=np.zeros(2)
ab=np.ones(4)
print(a)
print(ab)
```

```
[0. 0.]
[1. 1. 1. 1.]
```

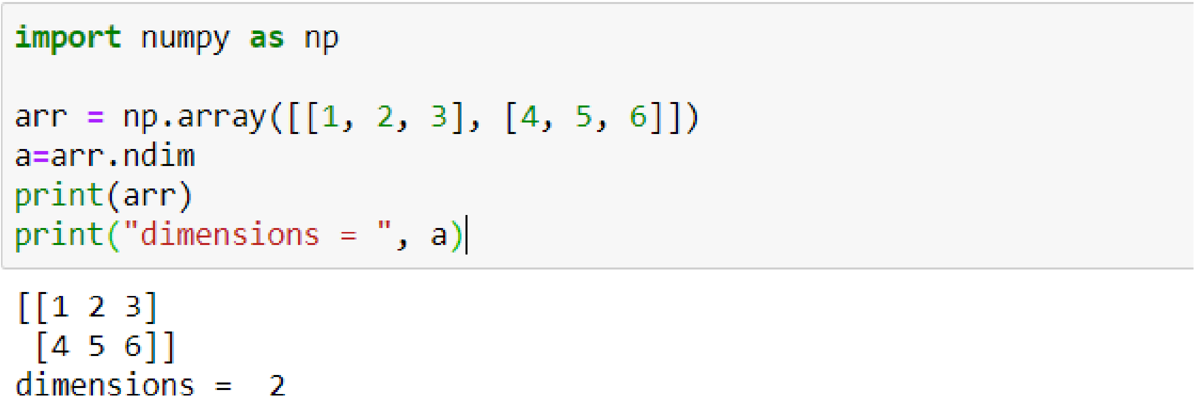
Program 3: Program to show function of delete and sort function.



```
a = np.array([1, 2, 3, 4, 5, 6, 7, 8])
print(a)


a=np.delete(a, 1)
print(a)
a = np.array([19, 22, 34, 14, 55, 76, 47, 8])
print(a)


a=np.sort(a)
print(a)
```

```
[1 2 3 4 5 6 7 8]
[1 3 4 5 6 7 8]
[19 22 34 14 55 76 47  8]
[ 8 14 19 22 34 47 55 76]
```

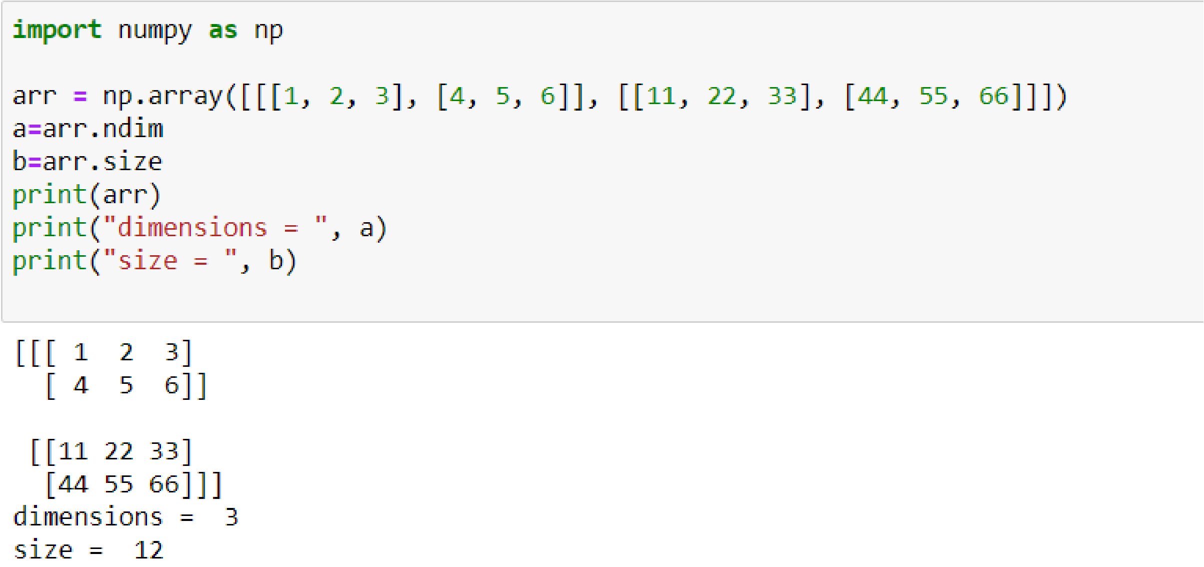
Program 4: Program to show the dimension of a 2d array.



```
import numpy as np

arr = np.array([[1, 2, 3], [4, 5, 6]])
a=arr.ndim
print(arr)
print("dimensions = ", a)
```

```
[[1 2 3]
 [4 5 6]]
dimensions =  2
```

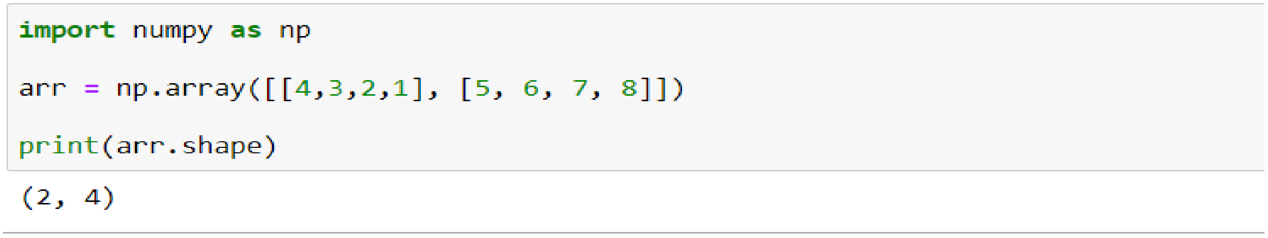
Program 5: Program to show the working of size function.



```
import numpy as np

arr = np.array([[[1, 2, 3], [4, 5, 6]], [[11, 22, 33], [44, 55, 66]]])
a=arr.ndim
b=arr.size
print(arr)
print("dimensions = ", a)
print("size = ", b)
```

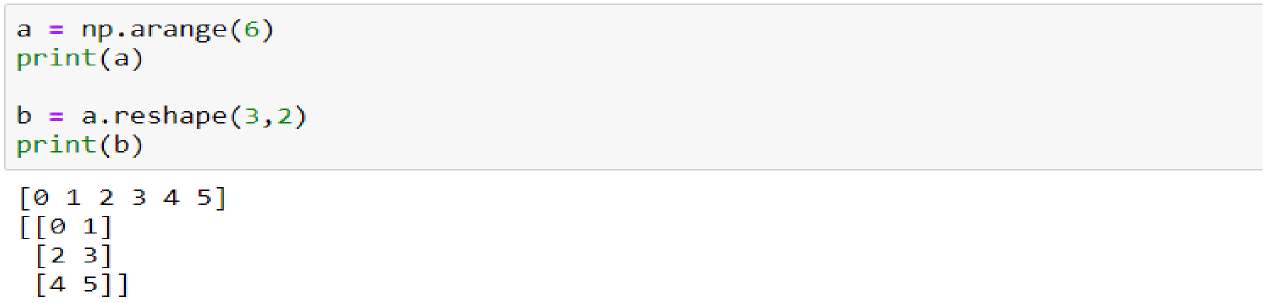
```
[[[ 1  2  3]
  [ 4  5  6]]

 [[11 22 33]
  [44 55 66]]]
dimensions =  3
size =  12
```

Program 6: Shape and Reshape function.



```
import numpy as np

arr = np.array([[4,3,2,1], [5, 6, 7, 8]])

print(arr.shape)
```

```
(2, 4)
```

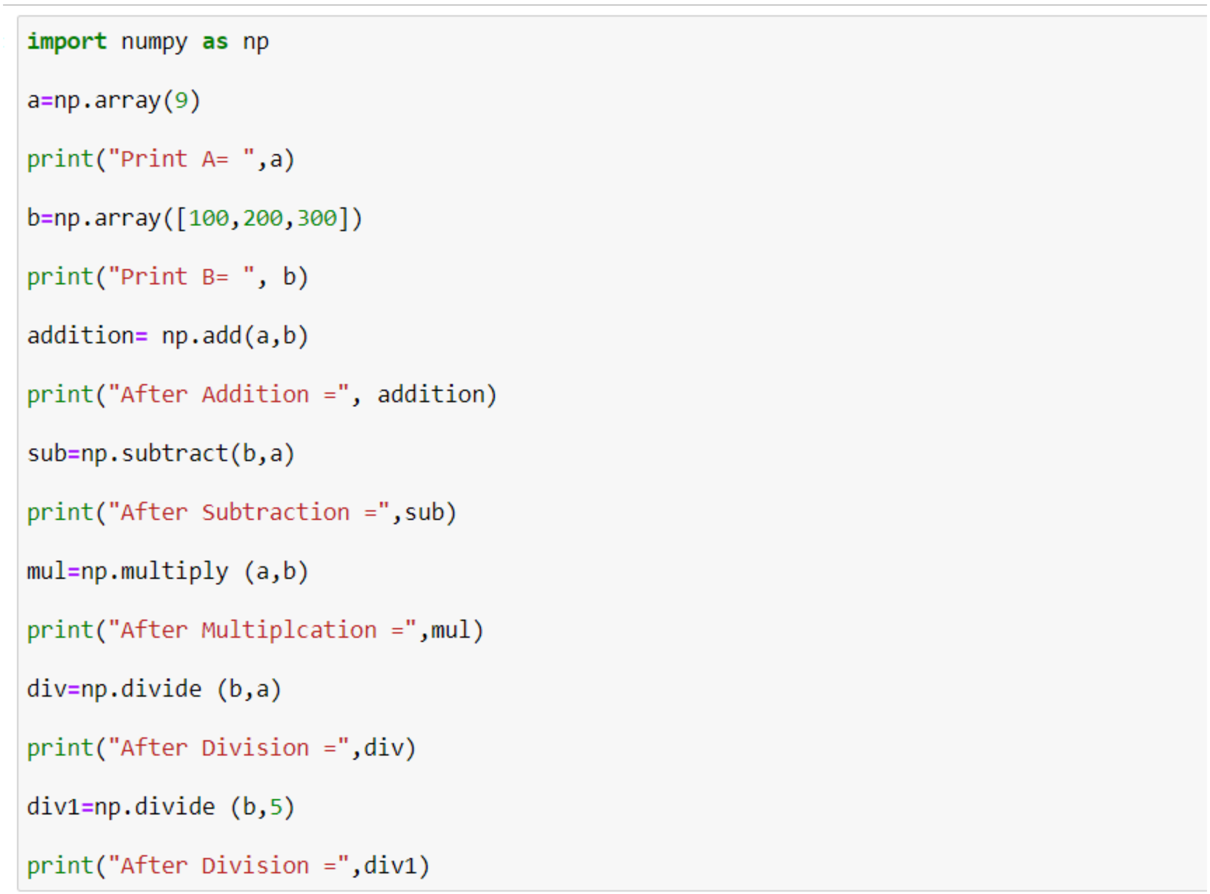


```
a = np.arange(6)
print(a)

b = a.reshape(3,2)
print(b)
```

```
[0 1 2 3 4 5]
[[0 1]
 [2 3]
 [4 5]]
```

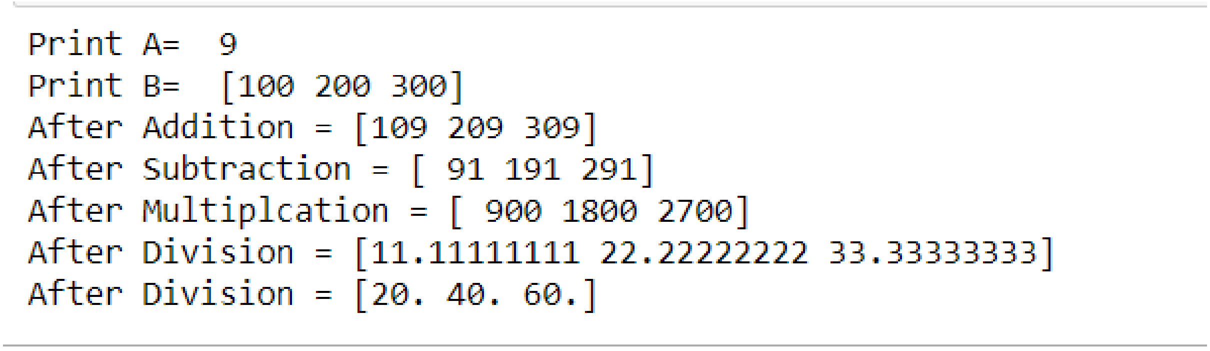
2. Numpy Arithmetic operations.

Program 1: Perform arithmetic operations using Numpy on array.



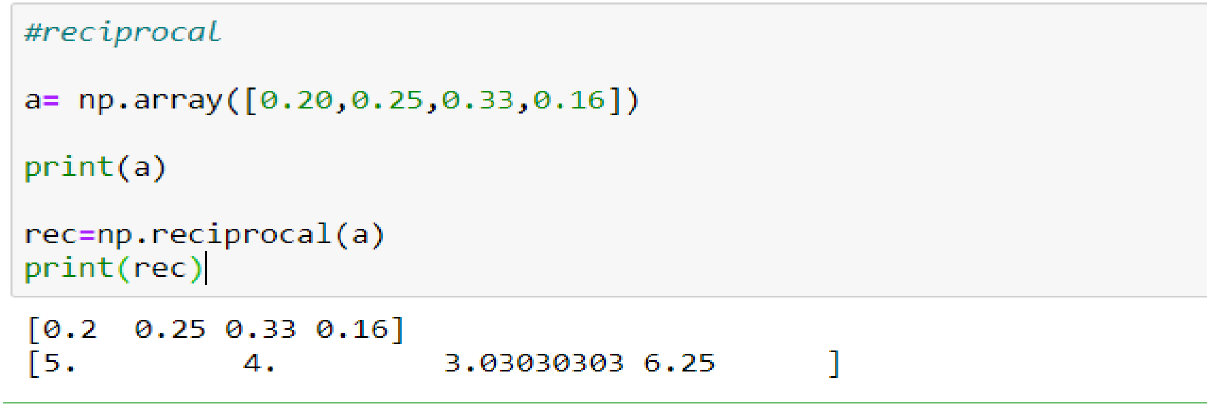
```
import numpy as np

a=np.array(9)

print("Print A= ",a)

b=np.array([100,200,300])

print("Print B= ", b)

addition= np.add(a,b)

print("After Addition =", addition)

sub=np.subtract(b,a)

print("After Subtraction =",sub)

mul=np.multiply (a,b)

print("After Multiplcation =",mul)

div=np.divide (b,a)

print("After Division =",div)

div1=np.divide (b,5)

print("After Division =",div1)
```

Output:



```
Print A=  9
Print B=  [100 200 300]
After Addition = [109 209 309]
After Subtraction = [ 91 191 291]
After Multiplcation = [ 900 1800 2700]
After Division = [11.11111111 22.22222222 33.33333333]
After Division = [20. 40. 60.]
```

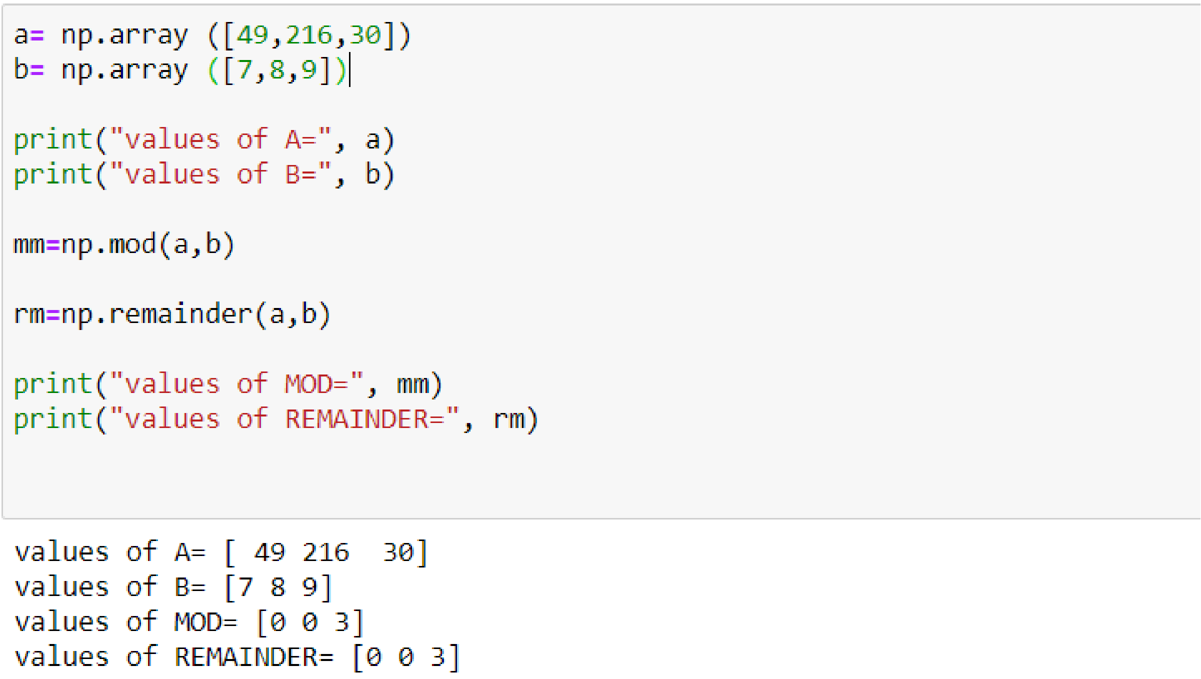
Program 2: Find the reciprocal of the array elements.



```
#reciprocal

a= np.array([0.20,0.25,0.33,0.16])

print(a)

rec=np.reciprocal(a)
print(rec)
```

```
[0.2  0.25 0.33 0.16]
[5.         4.         3.03030303 6.25      ]
```

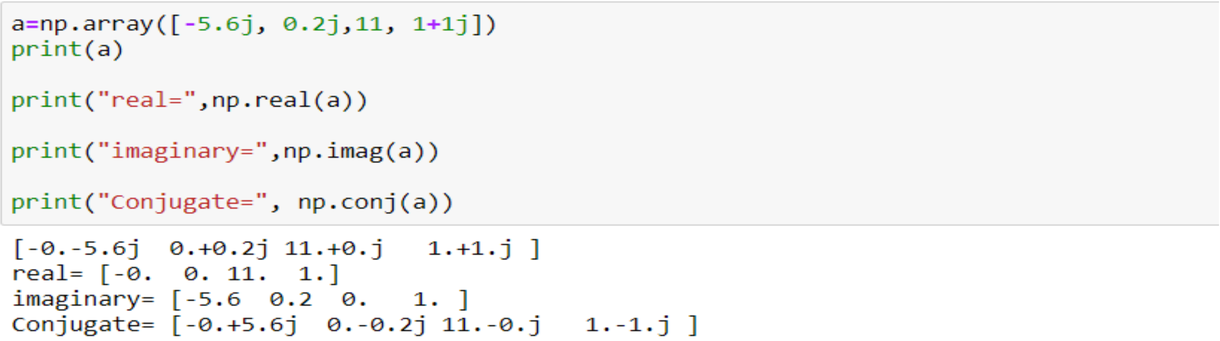
Program 3: Find the remainder values of the array elements when divided by other array elements.



```
a= np.array ([49,216,30])
b= np.array ([7,8,9])

print("values of A=", a)
print("values of B=", b)

mm=np.mod(a,b)

rm=np.remainder(a,b)

print("values of MOD=", mm)
print("values of REMAINDER=", rm)
```

```
values of A= [ 49 216  30]
values of B= [7 8 9]
values of MOD= [0 0 3]
values of REMAINDER= [0 0 3]
```

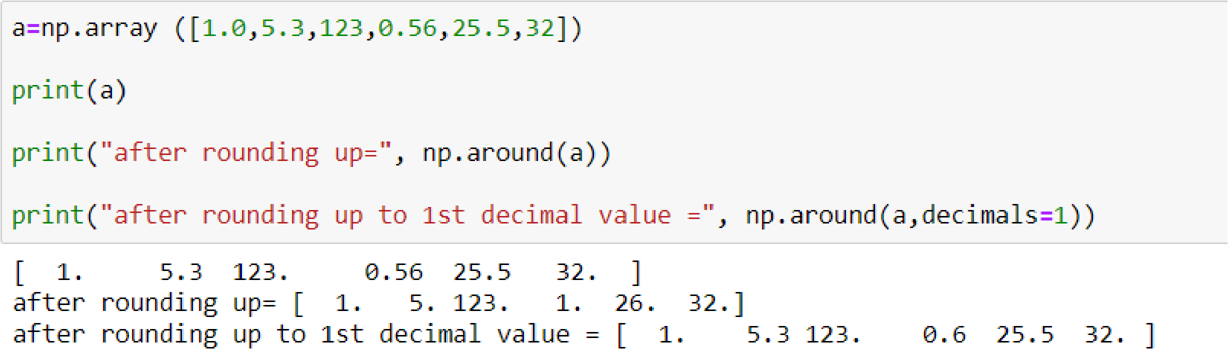
Program 4: Find real, imaginary and conjugate values.



```
a=np.array([-5.6j, 0.2j,11, 1+1j])
print(a)

print("real=",np.real(a))

print("imaginary=",np.imag(a))

print("Conjugate=", np.conj(a))
```

```
[-0.-5.6j  0.+0.2j 11.+0.j   1.+1.j ]
real= [-0.  0. 11.  1.]
imaginary= [-5.6  0.2  0.   1. ]
Conjugate= [-0.+5.6j  0.-0.2j 11.-0.j   1.-1.j ]
```

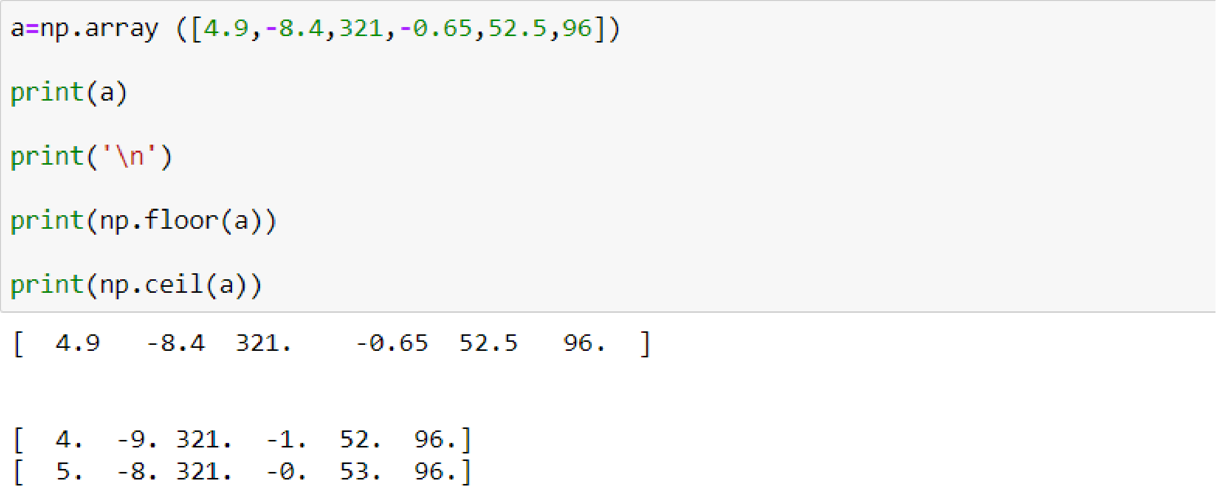
Program 5: Rounding up the array elements.



```
a=np.array ([1.0,5.3,123,0.56,25.5,32])

print(a)

print("after rounding up=", np.around(a))

print("after rounding up to 1st decimal value =", np.around(a,decimals=1))
```

```
[  1.      5.3   123.     0.56   25.5    32.  ]
after rounding up= [  1.    5. 123.    1.   26.   32.]
after rounding up to 1st decimal value = [  1.     5.3 123.     0.6   25.5   32. ]
```

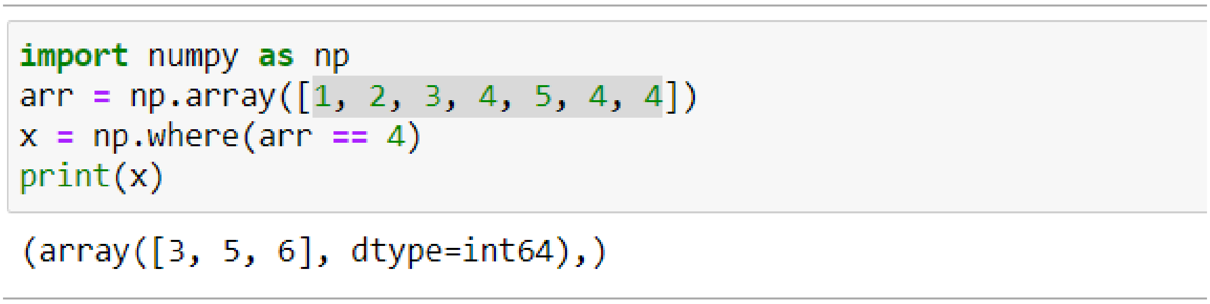
Program 6: floor and ceil function.



```
a=np.array ([4.9,-8.4,321,-0.65,52.5,96])

print(a)

print('\n')

print(np.floor(a))

print(np.ceil(a))
```

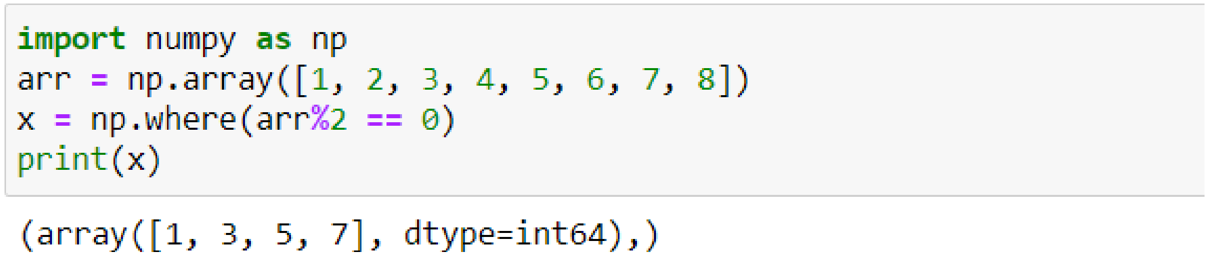
```
[  4.9   -8.4   321.    -0.65   52.5    96.   ]


[  4.  -9. 321.  -1.  52.  96.]
[  5.  -8. 321.  -0.  53.  96.]
```

I. Numpy array search



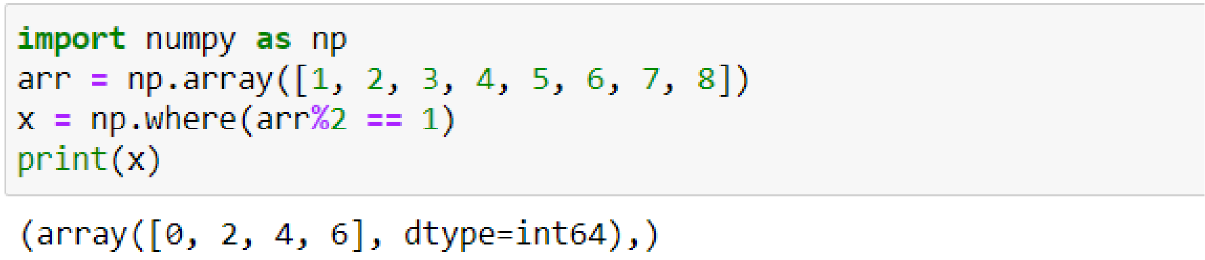
```
import numpy as np
arr = np.array([1, 2, 3, 4, 5, 4, 4])
x = np.where(arr == 4)
print(x)
```

```
(array([3, 5, 6], dtype=int64),)
```



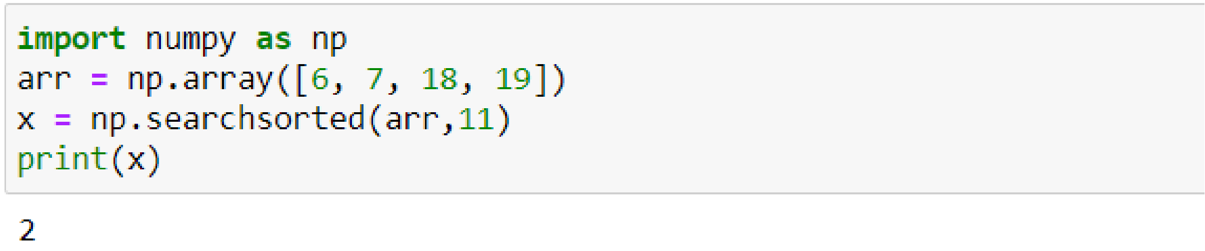
```
import numpy as np
arr = np.array([1, 2, 3, 4, 5, 6, 7, 8])
x = np.where(arr%2 == 0)
print(x)
```

```
(array([1, 3, 5, 7], dtype=int64),)
```



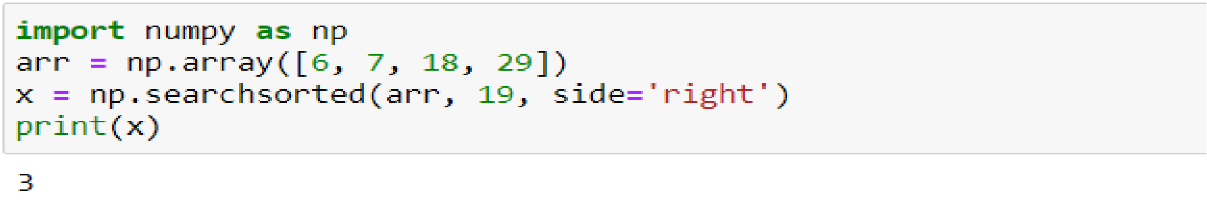
```
import numpy as np
arr = np.array([1, 2, 3, 4, 5, 6, 7, 8])
x = np.where(arr%2 == 1)
print(x)
```

```
(array([0, 2, 4, 6], dtype=int64),)
```



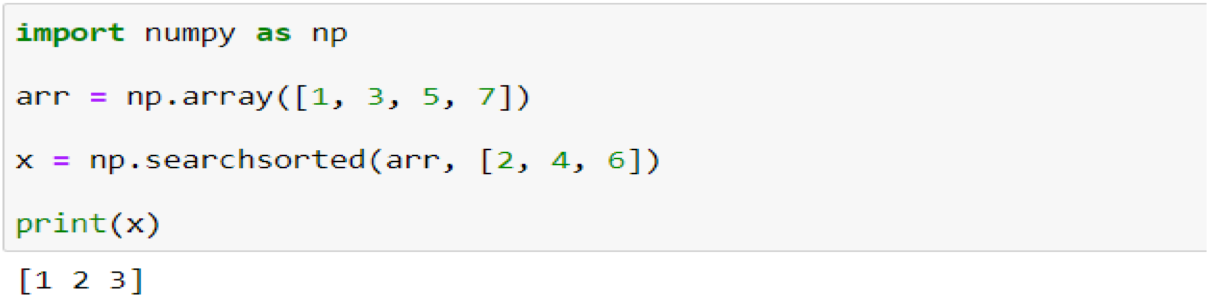
```
import numpy as np
arr = np.array([6, 7, 18, 19])
x = np.searchsorted(arr,11)
print(x)
```

```
2
```



```
import numpy as np
arr = np.array([6, 7, 18, 29])
x = np.searchsorted(arr, 19, side='right')
print(x)
```

```
3
```



```
import numpy as np

arr = np.array([1, 3, 5, 7])

x = np.searchsorted(arr, [2, 4, 6])

print(x)
```

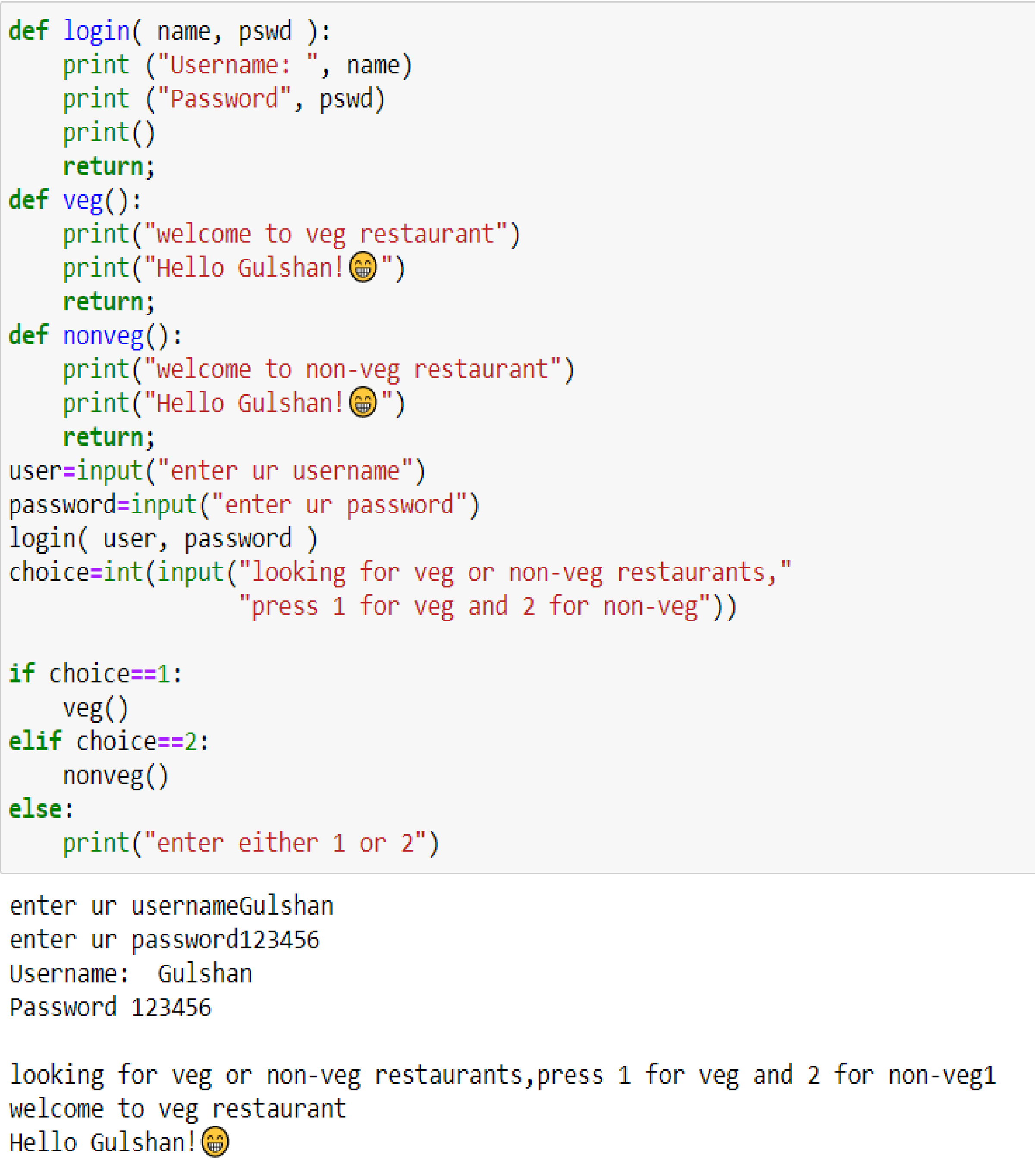
```
[1 2 3]
```

J. Restaurant

Objective: To choose a restaurant

Outcome: Select either between Veg or Non-Veg restaurant

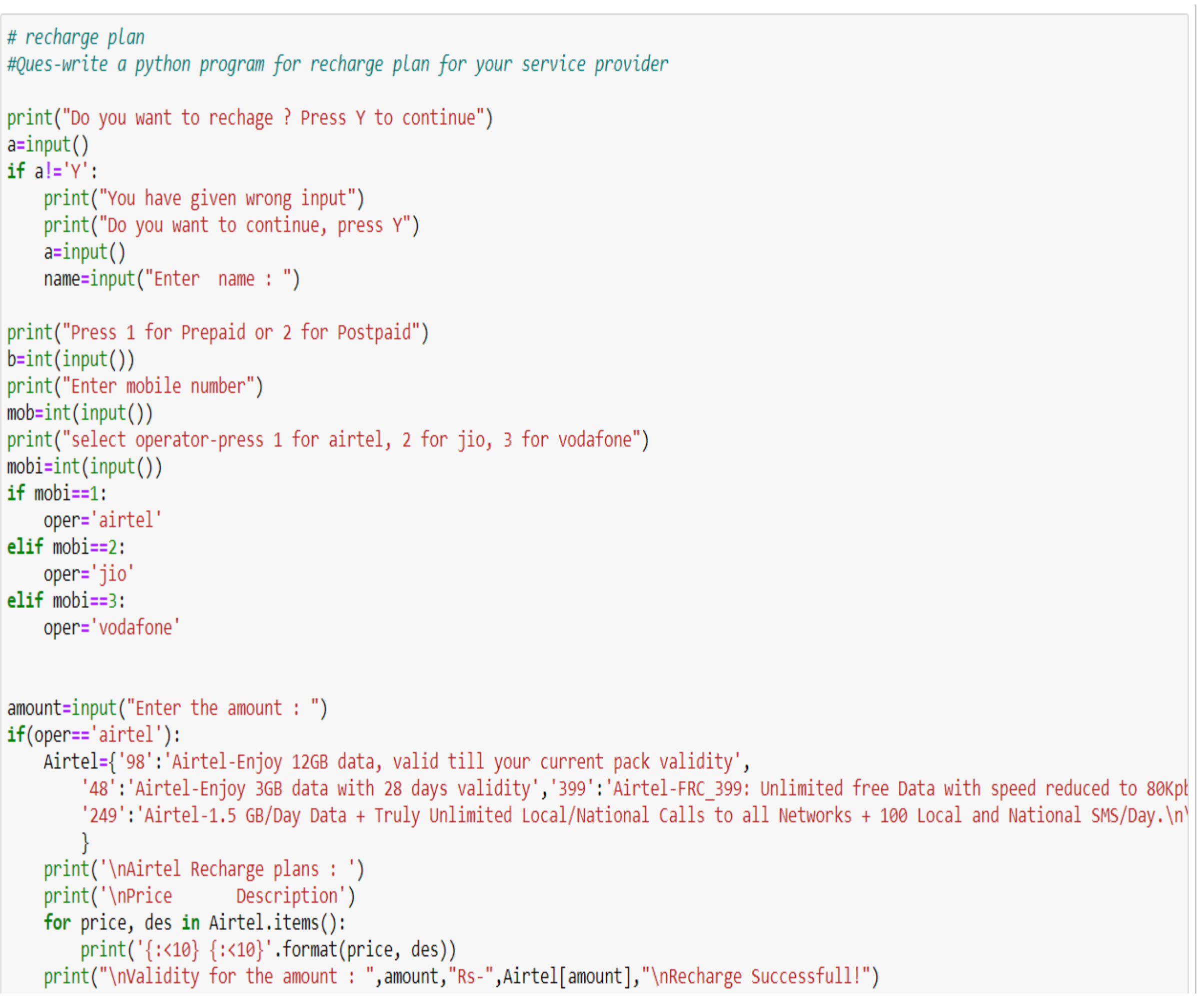
Conclusion: Successfully implemented Restaurant choice



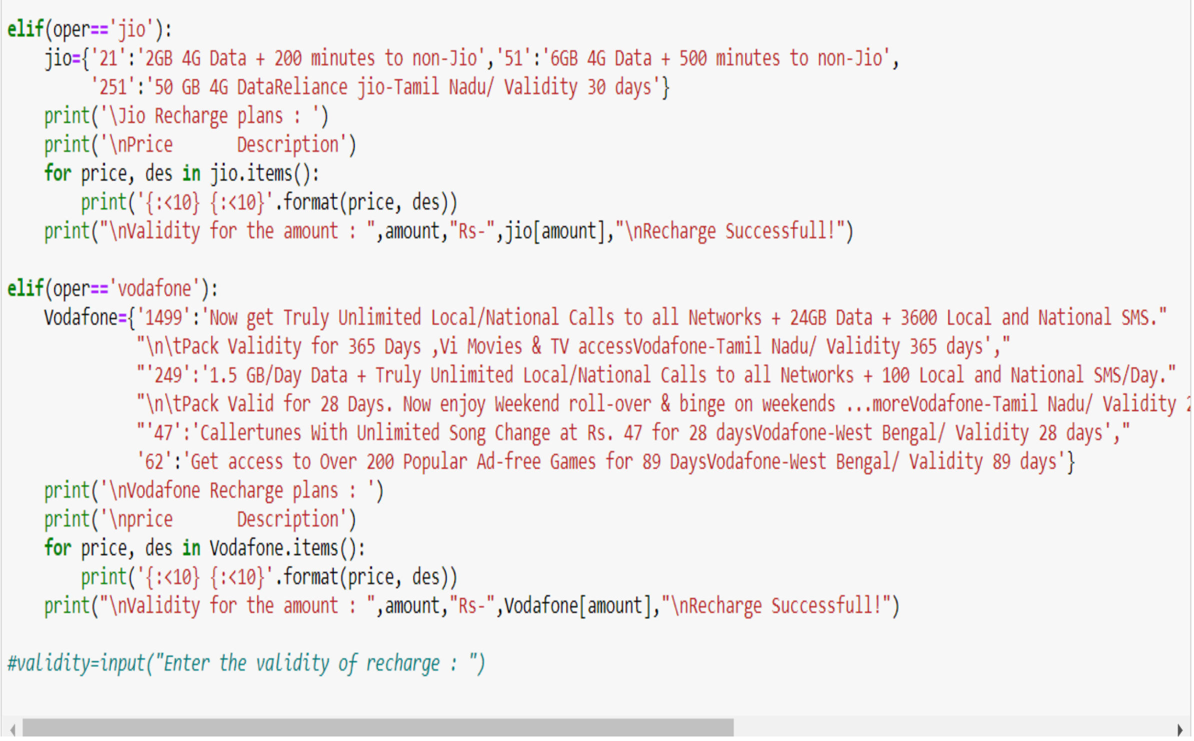
```
def login( name, pswd ):
    print ("Username: ", name)
    print ("Password", pswd)
    print()
    return;
def veg():
    print("welcome to veg restaurant")
    print("Hello Gulshan!😁")
    return;
def nonveg():
    print("welcome to non-veg restaurant")
    print("Hello Gulshan!😁")
    return;
user=input("enter ur username")
password=input("enter ur password")
login( user, password )
choice=int(input("looking for veg or non-veg restaurants,"
                 "press 1 for veg and 2 for non-veg"))

if choice==1:
    veg()
elif choice==2:
    nonveg()
else:
    print("enter either 1 or 2")
```

```
enter ur usernameGulshan
enter ur password123456
Username:  Gulshan
Password 123456

looking for veg or non-veg restaurants,press 1 for veg and 2 for non-veg1
welcome to veg restaurant
Hello Gulshan!😁
```

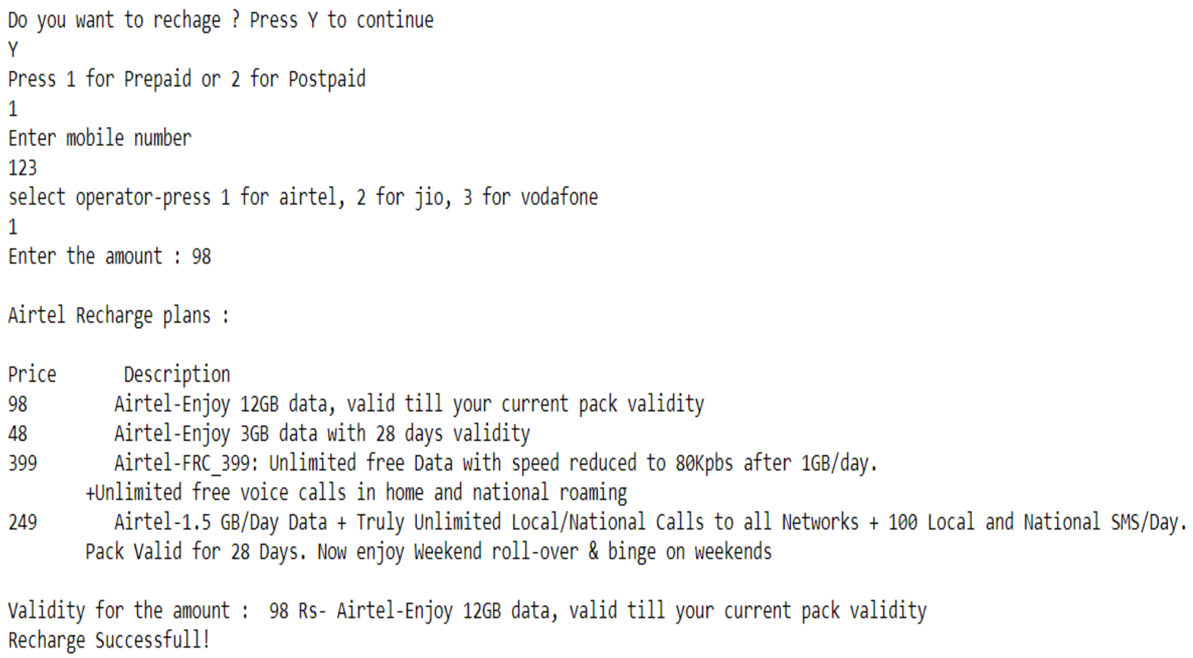
K. Recharge plan



```
# recharge plan
#Ques-write a python program for recharge plan for your service provider

print("Do you want to rechage ? Press Y to continue")
a=input()
if a!='Y':
    print("You have given wrong input")
    print("Do you want to continue, press Y")
    a=input()
    name=input("Enter  name : ")

print("Press 1 for Prepaid or 2 for Postpaid")
b=int(input())
print("Enter mobile number")
mob=int(input())
print("select operator-press 1 for airtel, 2 for jio, 3 for vodafone")
mobi=int(input())
if mobi==1:
    oper='airtel'
elif mobi==2:
    oper='jio'
elif mobi==3:
    oper='vodafone'


amount=input("Enter the amount : ")
if(oper=='airtel'):
    Airtel={'98':'Airtel-Enjoy 12GB data, valid till your current pack validity',
        '48':'Airtel-Enjoy 3GB data with 28 days validity','399':'Airtel-FRC_399: Unlimited free Data with speed reduced to 80Kpb
        '249':'Airtel-1.5 GB/Day Data + Truly Unlimited Local/National Calls to all Networks + 100 Local and National SMS/Day.\n\
        }
    print('\nAirtel Recharge plans : ')
    print('\nPrice       Description')
    for price, des in Airtel.items():
        print('{:<10} {:<10}'.format(price, des))
    print("\nValidity for the amount : ",amount,"Rs-",Airtel[amount],"\nRecharge Successfull!")
```



```
elif(oper=='jio'):
    jio={'21':'2GB 4G Data + 200 minutes to non-Jio','51':'6GB 4G Data + 500 minutes to non-Jio',
        '251':'50 GB 4G DataReliance jio-Tamil Nadu/ Validity 30 days'}
    print('\Jio Recharge plans : ')
    print('\nPrice       Description')
    for price, des in jio.items():
        print('{:<10} {:<10}'.format(price, des))
    print("\nValidity for the amount : ",amount,"Rs-",jio[amount],"\nRecharge Successfull!")

elif(oper=='vodafone'):
    Vodafone={'1499':'Now get Truly Unlimited Local/National Calls to all Networks + 24GB Data + 3600 Local and National SMS."
              "\n\tPack Validity for 365 Days ,Vi Movies & TV accessVodafone-Tamil Nadu/ Validity 365 days',"
              "'249':'1.5 GB/Day Data + Truly Unlimited Local/National Calls to all Networks + 100 Local and National SMS/Day."
              "\n\tPack Valid for 28 Days. Now enjoy Weekend roll-over & binge on weekends ...moreVodafone-Tamil Nadu/ Validity 2
              "'47':'Callertunes With Unlimited Song Change at Rs. 47 for 28 daysVodafone-West Bengal/ Validity 28 days',"
              '62':'Get access to Over 200 Popular Ad-free Games for 89 DaysVodafone-West Bengal/ Validity 89 days'}
    print('\nVodafone Recharge plans : ')
    print('\nprice       Description')
    for price, des in Vodafone.items():
        print('{:<10} {:<10}'.format(price, des))
    print("\nValidity for the amount : ",amount,"Rs-",Vodafone[amount],"\nRecharge Successfull!")

#validity=input("Enter the validity of recharge : ")
```



```
Do you want to rechage ? Press Y to continue
Y
Press 1 for Prepaid or 2 for Postpaid
1
Enter mobile number
123
select operator-press 1 for airtel, 2 for jio, 3 for vodafone
1
Enter the amount : 98

Airtel Recharge plans :

Price       Description
98         Airtel-Enjoy 12GB data, valid till your current pack validity
48         Airtel-Enjoy 3GB data with 28 days validity
399        Airtel-FRC_399: Unlimited free Data with speed reduced to 80Kpbs after 1GB/day.
        +Unlimited free voice calls in home and national roaming
249        Airtel-1.5 GB/Day Data + Truly Unlimited Local/National Calls to all Networks + 100 Local and National SMS/Day.
        Pack Valid for 28 Days. Now enjoy Weekend roll-over & binge on weekends

Validity for the amount :  98 Rs- Airtel-Enjoy 12GB data, valid till your current pack validity
Recharge Successfull!
```