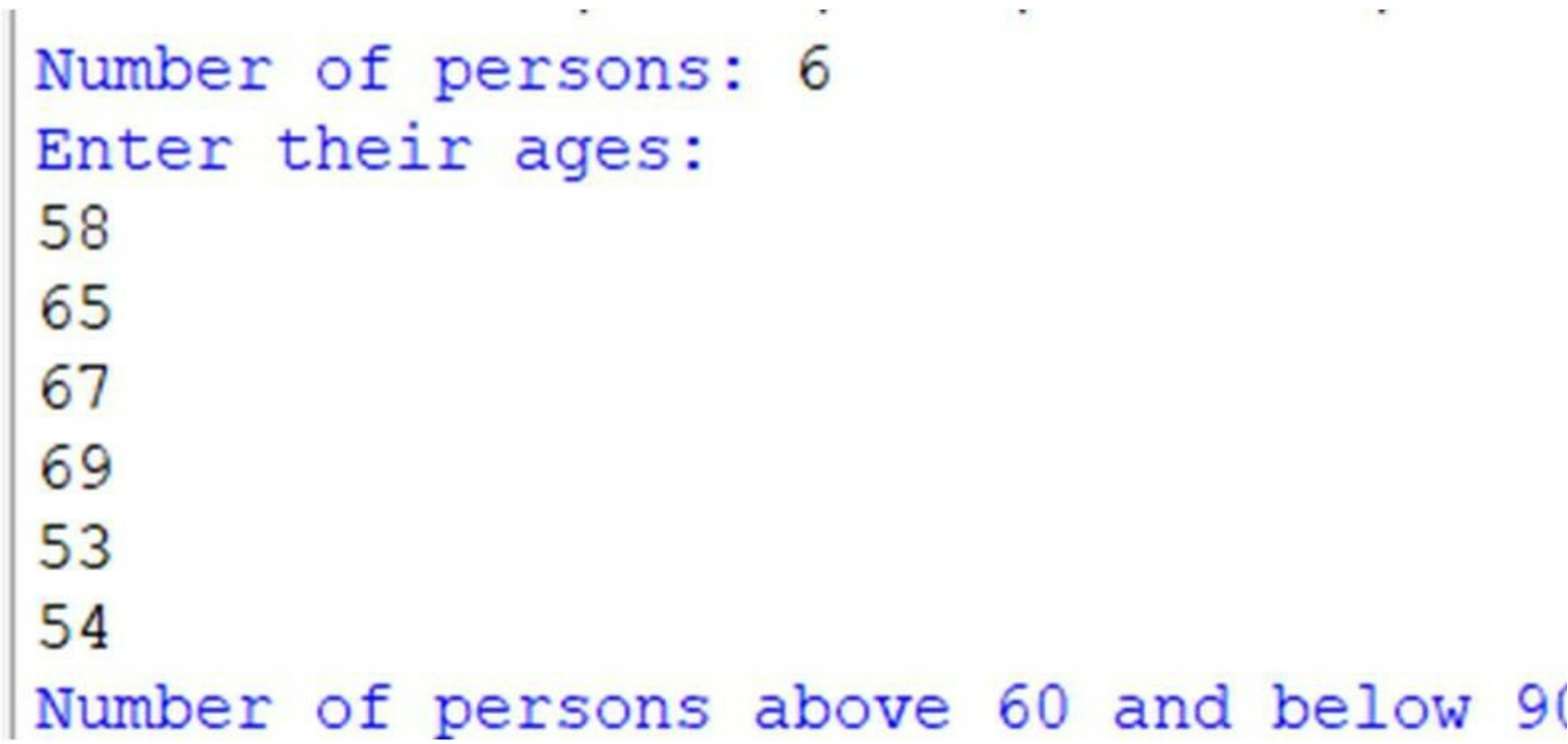
Count number of persons of age above 60 and below 90.

```
n = int(input("Number of persons: ")) print("Enter their ages:")

Ages = []

for i in range(n): Ages.append(int(input()))

count = 0

for x in Ages:

if (x >= 60 and x <= 90): count = count + 1

print("Number of persons above 60 and below 90 are:", count)
```

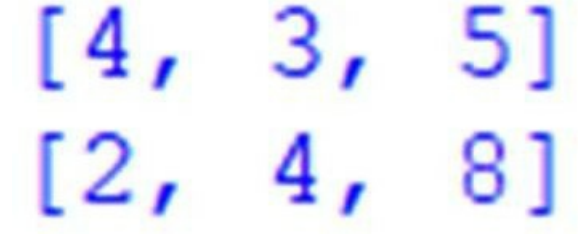
Output



Compute transpose of a matrix

```
x = [[4, 2],

[3, 4],

[5, 8]]

y = [[0, 0, 0],

[0, 0, 0]]

for i in range(len(x)):

for j in range(len(x[0])):

y[j][i] = x[i][j] for a in y:

print(a)
```

Output



Perform following operations on two matrices.

1. Addition

Subtraction

1. Multiplication
2. **Multiplication** x=[[5,4,0], [4,8,2],

[9,7,3]]

y=[[3,9,4],

[5,16,1],

[2,20,9]]

z=[[0,0,0],

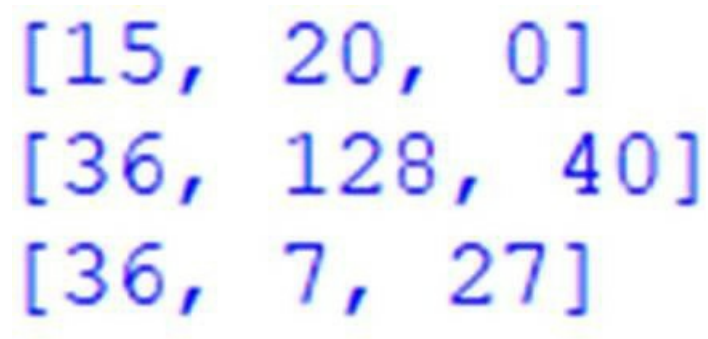
[0,0,0],

[0,0,0]]

for i in range(len(x)):

for j in range(len(x[0])): z[i][j]=x[i][j]*y[j][i] for a in z: print(a)

Output



1. **Addition** x=[[4,2], [3,4],

[5,8],

[9,0]]

y=[[0,7],

[0,0],

[5,7],

[4,9]]

z=[[0,0],

[0,0],

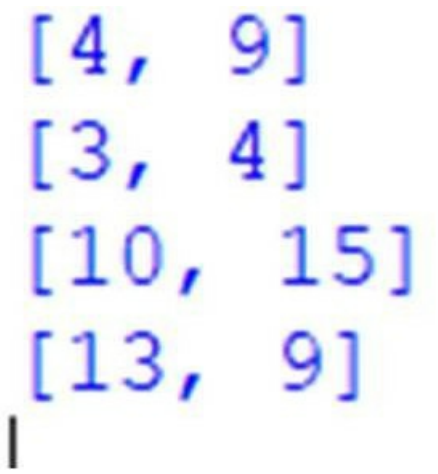
[0,0],

[0,0]]

for i in range(len(x)):

for j in range(len(x[0])): z[i][j]=x[i][j]+y[i][j] for a in z: print(a)

Output



1. **Subtraction**

x=[[4,2],

[3,4],

[5,8],

[9,0]]

y=[[0,7],

[0,0],

[5,7],

[4,9]]

z=[[0,0],

[0,0],

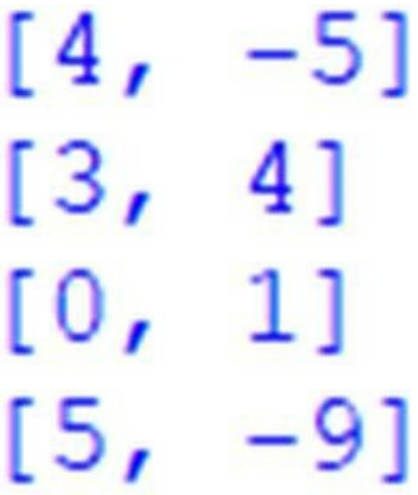
[0,0],

[0,0]]

for i in range(len(x)):

for j in range(len(x[0])): z[i][j]=x[i][j]-y[i][j] for a in z: print(a)

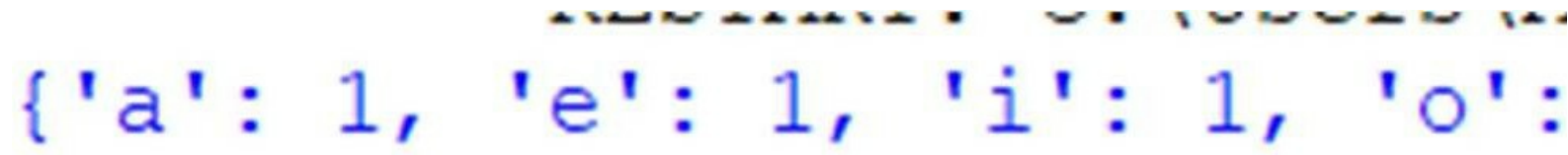
Output



Count occurrence of vowels.

```
def Check_Vow(string, vowels): string = string.casefold()

count = {}.fromkeys(vowels, 0) for character in string:

if character in count: count[character] += 1

return count vowels = 'aeiou'

string = "Hey, what's going on" result = Check_Vow(string, vowels) print(result)
```

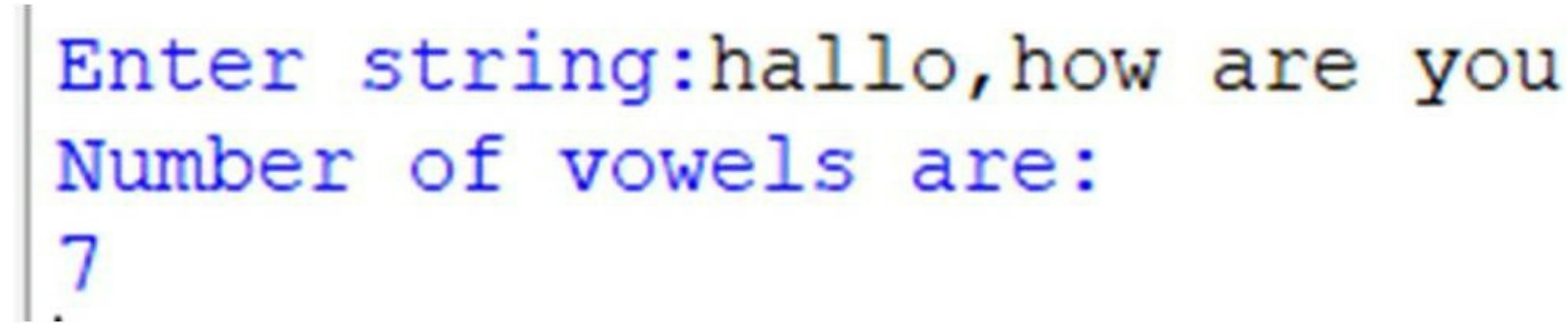
Output



Count total number of vowels in a word.

```
string = input("Enter string:") vowels = 0

for i in string:

if i in 'aeiouAEIOU': vowels = vowels + 1

print("Number of vowels are:") print(vowels)
```

Output



1. Determine whether a string is palindrome or not.

```
a = "Ratan" reverse = a[::-1] if a == reverse:

print("string is a palindrome") else:

print("string is not a palindrome")
```

Output

Perform following operations on a list of numbers: 1) Insert an element 2) delete an element 3) sort the list 4) delete entire list.

Insert an element

l= [50, 70, 20, 80, 56, 43]

l.insert(4, 'R') print(l)

output

```
[50, 70, 20, 80, 'R', 56, 43]
```

delete an element

l= [50, 70, 20, 80, 56, 43]

l.pop(2) print(l)

```
[50, 70, 80, 56, 43]
```
Output

sort the list

l=[50,90,20,65,96,43]

l.sort() print(l)

Output

```
[20, 43, 50, 65, 90, 96]
```

1. **delete entire list**

l=[50,90,20,65,96,43]

```
[]
```
l.clear() print(l) **Output**

Display word after Sorting in alphabetical order.

my_str = input ("Enter a string: ") words = my_str.split() words.sort()

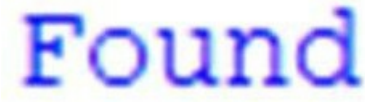
for word in words: print(word)

```
Enter a string: mango banana apple grap
apple
banana
grapes
mango
plum
```

Output

Perform sequential search on a list of given numbers.

```
def search(List, n):

for i in range(len(List)): if List[i] == n:

return True return False

List = [1, 2, 'harry', 4, 'geeks', 6] n = 'geeks'

if search(List, n): print("Found") else:

print("Not Found")
```

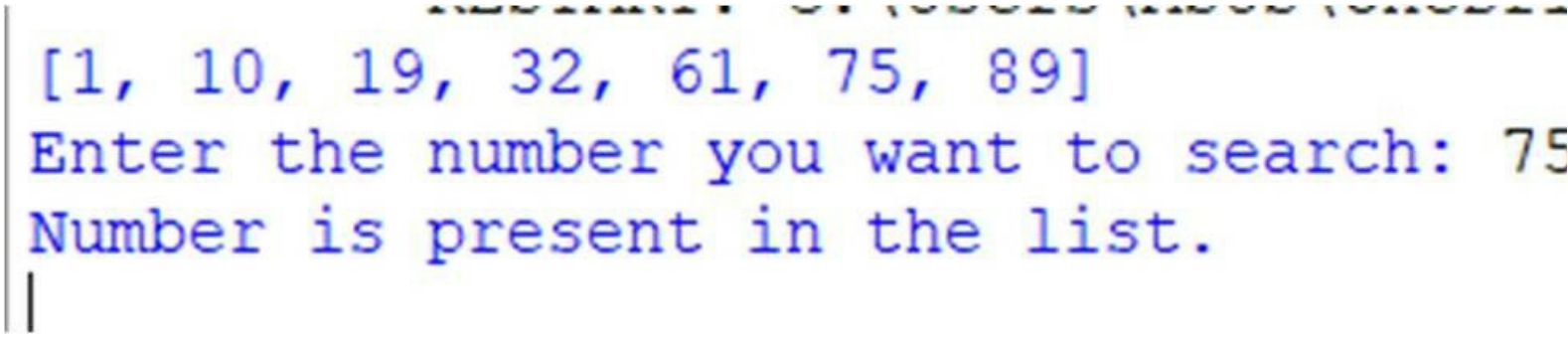


Output

1. Perform sequential search on ordered list of given numbers.

```
l= [1, 32, 89, 75, 61, 19, 10]

l.sort() print(l) count = 0

n = int(input("Enter the number you want to search: ")) for i in l:

if i == n: count = 1 break

if count == 1:

print("Number is present in the list.") else:

print("Number is not present in the list.")
```

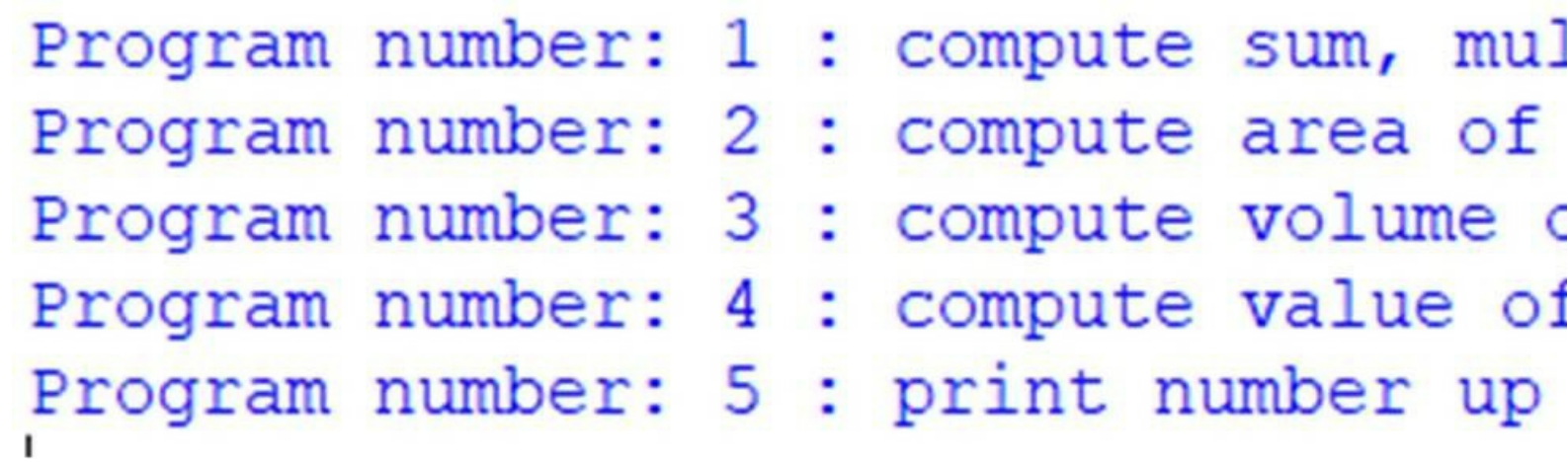
Output



Maintain practical note book as per their serial numbers in library using Python dictionary.

```
d1 = {}

d1[5] = "print number up to n 3,2,4,5" d1[1] = "compute sum, multiplication" d1[4] = "compute value of square"

d1[3] = "compute volume of following 3 shapes" d1[2] = "compute area of following shapes"

d = sorted(d1.items()) for i, j in d:

print("Program number:", i, ":", j)
```

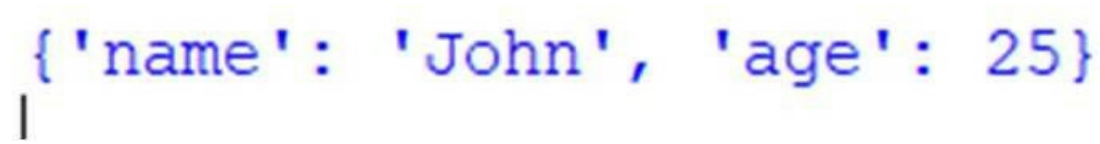
Output



1. Perform following operations on dictionary 1) Insert 2) delete 3) change
2. **Insert**

my_dict = {} my_dict['name'] = 'John' my_dict['age'] = 25 print(my_dict)

Output

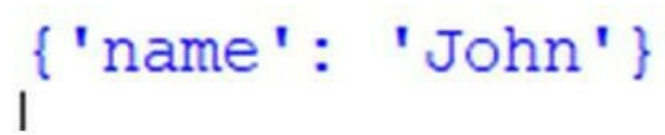


1. **delete**

my_dict = {'name': 'John', 'age': 25} del my_dict['age']

print(my_dict)

Output

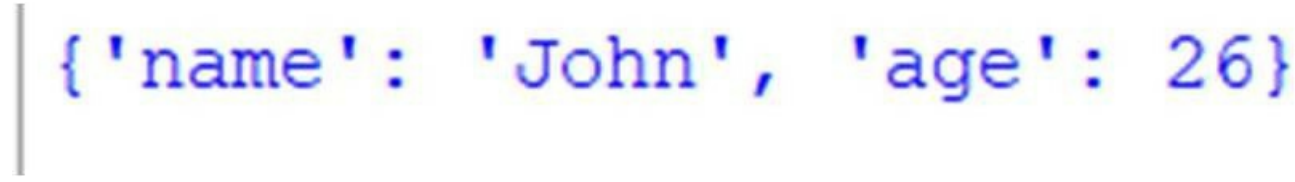


1. **change**

my_dict = {'name': 'John', 'age': 25} my_dict['age'] = 26

print(my_dict)

Output



Check whether a number is in a given range using functions.

def FUN(st, end, key): A = set()

for i in range(st, end + 1): A.add(i)

print("SET A is:", A)

if key in A:

print(key, "is present in A") else:

```
print(key, "is absent in A")

a = int(input("Enter the starting point of range: ")) b = int(input("Enter the end point of range: "))

n = int(input("Enter the element to find: ")) if a >= b:

print("Kindly enter the first value as smaller.") else:

FUN(a, b, n)
```

Output

```
Enter the starting point of range: 4
Enter the end point of range: 16
Enter the element to find: 9
SET A is: {4, 5, 6, 7, 8, 9, 10, 11, 12
9 is present in A
```

Write a Python function that accepts a string and calculates number of upper case letters and lower case letters available in that string

```
def check(string): upper = 0

lower = 0

for x in string: if x.isupper():

upper += 1 elif x.islower(): lower += 1

print("Uppercase letter =", upper) print("Lowercase letter =", lower)

y = input("Enter any string: ") check(y)
```

Output

```
Enter any string: Raj Convent Sch
Uppercase letter = 3
Lowercase letter = 13
```

To find the Max of three numbers using functions

```
def max(x, y, z):

if (x > y and x > z): return x

elif (y > x and y > z): return y

else:

return z
```

```
a = int(input("Enter First number: "))

b = int(input("Enter Second number: ")) c = int(input("Enter Third number: "))

result = max(a, b, c) print("Maximum Number is:", result)
```

Output