06 - Strings in Python

Ex. No.	**:**	**6.1**	**Date:**	**04.05.24**
Register No.:		**231901020**	**Name:**	**KAVIYA.V**

Count Chars

Write a python program to count all letters, digits, and special symbols respectively from a given string

CODING:

```
a=input()
b=0
c=0
d=0
for i in range (0,len(a)):
    if a[i].isalpha():
        b=b+1
    elif a[i].isdigit():
        c=c+1
    else:
        d=d+1
print(b)
print(c)
print(d)
```

	Input	Expected	Got	
✔	rec@123	3 3 1	3 3 1	✔
✔	P@#yn26at^&i5ve	8 3 4	8 3 4	✔
✔	abc@12&	3 2 2	3 2 2	✔

For example:

Input Result
rec@123
3
3
1

Ex. No. : 6.2 **Date: 04.05.24**

Register No.: 231901020 **Name: KAVIYA.V**

Decompress the String

Assume that the given string has enough memory. Don't use any extra space(IN-PLACE)

CODING:

```
s=input()
t=0
char=''
for i in s:
    if i.isalpha():
        print(char*t,end='')
        t=0
        char=i
    else:
        t=t*10+int(i)
print(char*t,end='')
```

	Input	Expected	Got	
✔	a2b4c6	aabbbbccccccc	aabbbbccccccc	✔
✔	a12b3d4	aaaaaaaaaaaabbbddddd	aaaaaaaaaaaabbbddddd	✔

Sample Input 1
a2b4c6

Sample Output 1
aabbbbccccccc

Ex. No. : 6.3 **Date: 04.05.24**

Register No.: 231901020 **Name: KAVIYA.V**

First N Common Chars

Two string values S1, S2 are passed as the input. The program must print first N characters present in S1 which are also present in S2.

CODING:

```
a=input()
b=input()
n=int(input())
c=""
for i in a:
   if i in b and i not in c:
      c+=i
      if len(c)==n:
         break
print(c)
```

Input Format:

The first line contains S1.
The second line contains S2.
The third line contains N.

Output Format:

The first line contains the N characters present in S1 which are also present in S2.

Boundary Conditions:

2 <= N <= 10
2 <= Length of S1, S2 <= 1000

Example Input/Output 1:

Input:

abcbde
cdefghbb
3

Output:

bcd

Note:

b occurs twice in common but must be printed only once.

Ex. No. : 6.4 **Date: 04.05.24**

Register No.: 231901020 **Name: KAVIYA.V**

Remove Characters

Given two Strings s1 and s2, remove all the characters from s1 which is present in s2.

Constraints
1<= string length <= 200

CODING:

```
a=input()
b=input()
c=list(a)
d=list(b)
e=[]
for i in range(len(c)):
    if(c[i] not in d):
        e.append(c[i])
s=""
for j in range(len(e)):
    s+=e[j]
print(s)
```

	Input	Expected	Got	
✔	experience enc	xpri	xpri	✔

Sample Input 1
experience
enc

Sample Output 1
xpri

EXP. No. : 6.5 **Date: 04.05.24**

Register No.: 231901020 **Name: KAVIYA.V**

Remove Palindrome Words

String should contain only the words are not palindrome.

Sample Input 1
Malayalam is my mother tongue

Sample Output 1
is my mother tongue

CODING:

```
a=input()
l=a.lower()
b=l.split(' ')
c=[]
for i in range(len(b)):
    if b[i][::1]!=b[i][::-1]:
        c.append(b[i])
for i in c:
    print(i,end=' ')
```

For example:

Input	Expected
Malayalam is my mother tongue	is my mother tongue
He did a good deed	he good

	Input	Expected	Got	
✔	Malayalam is my mother tongue	is my mother tongue	is my mother tongue	✔

Ex. No. : 6.6 **Date: 04.05.24**

Register No.: 231901020 **Name: KAVIYA.V**

Return Second World in Uppercase

Write a program that takes as input a string (sentence), and returns its second word in uppercase.

For example:

If input is “Wipro Technologies Bangalore” the function should return “TECHNOLOGIES”
If input is “Hello World” the function should return “WORLD”
If input is “Hello” the program should return “LESS”

NOTE 1: If input is a sentence with less than 2 words, the program should return the word “LESS”.
NOTE 2: The result should have no leading or trailing spaces.

CODING:

```
s=input().split()
if(len(s)>=2):
   print(s[1].upper())
else:
   print("Less")
```

For example:

Input Result
Wipro Technologies Bangalore
TECHNOLOGIES
Hello World
WORLD
Hello
LESS

Ex. No. : 6.7 **Date: 04.05.24**

Register No.: 231901020 **Name: KAVIYA.V**

Revers String

Reverse a string without affecting special characters. Given a string S, containing special characters and all the alphabets, reverse the string without affecting the positions of the special characters.

CODING:

```
s=input().strip()
s=list(s)
l=0
r=len(s)-1
while l<r:
    if not s[l].isalpha():
        l+=1
    elif not s[r].isalpha():
        r-=1
    else:
        s[l],s[r]=s[r],s[l]
        l+=1
        r-=1
print(''.join(s))
```

	Input	Expected	Got	
✔	A&B	B&A	B&A	✔

fInput:
A&B
Output:
B&A
Explanation: As we ignore '&' and
As we ignore '&' and then reverse, so answer is "B&A".

For example:

Input Result
A&x#
x&A#

Ex. No. : 6.8 **Date: 04.05.24**

Register No.: 231901020 **Name: KAVIYA.V**

String characters balance Test

Write a program to check if two strings are balanced. For example, strings s1 and s2 are balanced if all the characters in the s1 are present in s2. The character's position doesn't matter. If balanced display as "true" ,otherwise "false".

CODING:

```
a=input()
b=input()
if b.find(a)==-1:
    print("False")
else:
    print("True")
```

	Input	Expected	Got	
✔	Yn PYnative	True	True	✔
✔	Ynf PYnative	False	False	✔

For example:

Input Result
Yn
PYnative
True

Ex. No. : 6.9 **Date: 04.05.24**

Register No.: 231901020 **Name: KAVIYA.V**

Unique Names

In this exercise, you will create a program that reads words from the user until the user enters a blank line. After the user enters a blank line your program should display each word entered by the user exactly once. The words should be displayed in the same order that they were first entered. For example, if the user enters:

CODING:

```
a=[]
for i in range(5):
    word=input()
    if word not in a:
        a.append(word)
for word in a:
    print(word)
```

Input:
first
second
first
third
second

then your program should display:

Output:
first
second
third

	Input	Expected	Got	
✔	first second first third second	first second third	first second third	✔
✔	rec cse it rec cse	rec cse it	rec cse it	✔

Ex. No. : 6.10 **Date: 04.05.24**

Register No.: 231901020 **Name: KAVIYA.V**

Username Domain Extension

Given a string S which is of the format USERNAME@DOMAIN.EXTENSION, the program must print the EXTENSION, DOMAIN, USERNAME in the reverse order.

Input Format:
The first line contains S.

Output Format:
The first line contains EXTENSION.
The second line contains DOMAIN.
The third line contains USERNAME.

Boundary Condition:
1 <= Length of S <= 100

CODING:

```
s=input()
a=s.split("@")
b=a[1].partition(".")
c=list(b)
print(b[-1])
print(b[0])
print(a[0])
```

	Input	Expected	Got	
✔	abcd@gmail.com	com gmail abcd	com gmail abcd	✔

Example Input/Output 1:
Input:
vijayakumar.r@rajalakshmi.edu.in
Output:

edu.in
rajalakshmi
vijayakumar.r