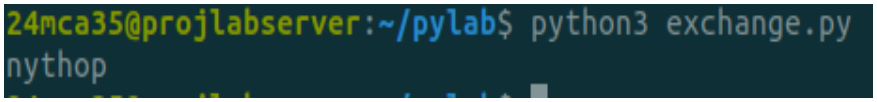
CYCLE 2

PROGRAMME 1

AIM:
Create a string from the given string where the first and last character are exchanged
Source Code:

```
string="python"
output=string[-1]+string[1:-1]+string[0]
print(output)
```

OUTPUT:

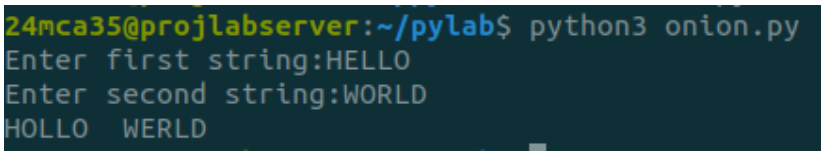


PROGRAMME 2

AIM:
Create a single string seperated with space from two strings by swapping the character at position 1
Source Code:

```
str1=input("Enter first string:")
str2=input("Enter second string:")
n1=len(str1)
n2=len(str2)
str1a=str1[1]
str2a=str2[1]
print(str1[0]+str2a+str1[2:n1+1],"",str2[0]+str1a+str2[2:])
```

OUTPUT:

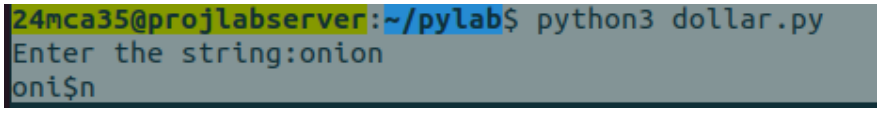


PROGRAMME 3

AIM:
Get a string from an input string where all occurences of the first character are replaced with ‘$’,except the first character
Source Code:

```
str1=input("Enter the string:")
char=str1[0]
str1=str1.replace(char,'$')
```

print(char+str1[1:])
OUTPUT:

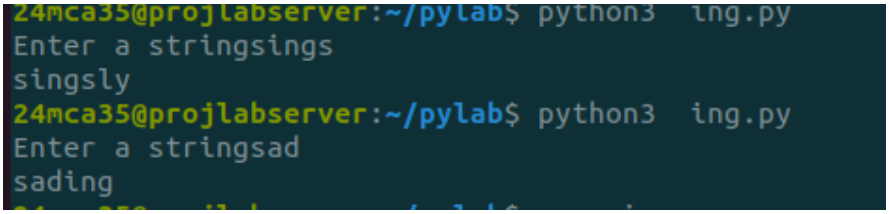


PROGRAMME 4

AIM:
Add ‘ing’ at the end of a given string.If it already ends with ‘ing’,then all ‘ly’
Source Code:

```
str1=input("Enter a string")
if "ing" in str1:
    print(str1+"ly")
else:
    print(str1+"ing")
```

OUTPUT:



PROGRAMME 5

AIM:
Write a python pgmm to read two lists color-list 1 and color-list 2 .print out all the colors from color-list 1 not contained in color-list 2
Source Code:

```
c1=input("Enter the first color:")
c2=input("Enter the second color:")
c3=input("Enter the third color:")
c4=input("Enter the fourth color:")
c5=input("Enter the fifth color:")
color_list1=[c1,c2,c3,c4,c5]
print("The first list of colours is:",color_list1)
co1=input("Enter first color:")
co2=input("Enter second color:")
co3=input("Enter third color:")
co4=input("Enter fourth color:")
co5=input("Enter fifth color:")
color_list2=[co1,co2,co3,co4,co5]
print("The second list of colors is:",color_list2)
result_list=[]
```

```
for i in color_list1:
    if i not in color_list2:
        result_list.append(i)
print("The colors from first list not contained in second list are:",result_list)
```

OUTPUT:

```
24mca35@projlabserver:~/pylab$ python3 color.py
Enter the first color:red
Enter the second color:green
Enter the third color:blue
Enter the fourth color:yellow
Enter the fifth color:pink
The first list of colours is: ['red', 'green', 'blue', 'yellow', 'pink']
Enter first color:red
Enter second color:black
Enter third color:pink
Enter fourth color:white
Enter fifth color:brown
The second list of colors is: ['red', 'black', 'pink', 'white', 'brown']
The colors from first list not contained in second list are: ['green', 'blue', 'yellow']
```

PROGRAMME 6

AIM:
Count the no.of.characters in a string
Source Code:

```
n=input("Enter the string:").lower()
s={}
for i in n:
    if i in s:
        s[i]+=1
    else:
        s[i]=1
print(s)
```

OUTPUT:

```
24mca35@projlabserver:~/pylab$ python3 cha.py
Enter the string:lekshmi r rajan
{'l': 1, 'e': 1, 'k': 1, 's': 1, 'h': 1, 'm': 1, 'i': 1, ' ': 2, 'r': 2, 'a': 2, 'j': 1, 'n': 1}
```

PROGRAMME 7

AIM:
Store a list of first name and count the occurences of ‘a’ within the lists
Source Code:

```
names=input("Enter first names seperated by commas:")
count_a=names.lower().count('a')
print(f"The letter 'a' appears {count_a} times in the list of names")
```

OUTPUT:

```
24mca35@projlabserver:~/pylab$ python3 occ.py
Enter first names seperated by commas:['a','d','d','c','a','b','b','a','c','d','e']
The letter 'a' appears 3 times in the list of names
```

PROGRAMME 9

AIM:
Writ a pgmm to prompt the user for a list of integers.For all value greater than 100.Store ‘over’ instead
Source Code:

```
input_numbers=input("Enter a list of integers(comma seperated):")
numbers=[int(num.strip()) for num in input_numbers.split(",")]
result=['over' if num>100 else num for num in numbers]
print("resulting list",result)
```

OUTPUT:

```
24mca35@projlabserver:~/pylab$ python3 over.py
Enter a list of integers(comma seperated):50,120,150,130,45
resulting list [50, 'over', 'over', 'over', 45]
```

PROGRAMME 10

AIM:
From a list of integers,create a list after removing even numbers
Source Code:

```
input_numbers=[10,21,32,43,54,65,76,87,98]
odd_numbers=[num for num in input_numbers if num % 2  !=0]
print("List after removing even numbers:",odd_numbers)
```

OUTPUT:

```
24mca35@projlabserver:~/pylab$ nano odd.py
24mca35@projlabserver:~/pylab$ python3 odd.py
List after removing even numbers: [21, 43, 65, 87]
```

PROGRAMME 11

AIM:

Accept a list of words and return the length of the longest word
Source Code:

```
input_words=input("Enter a list of words(seperated  by spaces):")
words=input_words.split()
if words:
    longest_word_length=max(len(word) for word in words)
else:
    longest_word_length=a
print("length of the lomgestword:",longest_word_length)
```

OUTPUT:

```
24mca35@projlabserver:~/pylab$ python3 word.py
Enter a list of words(seperated  by spaces):sky blue red pic maths
length of the lomgestword: 5
```

PROGRAMME 12

AIM:
Write a program to prompt the user to enter two lists of integers and check (
a) Whether lists are of the same length.
(b) Whether the list sums to the same value.
(c) Whether any value occurs in both Lists.
Source Code:

```
list1_input=input("Enter the first list of integers(comma seperated):")
list1=[int(num.strip()) for num in list1_input.split(",")]
list2_input=input("Enter the second list of integers(comma seperated):")
list2=[int(num.strip()) for num in list2_input.split(",")]
same_length=len(list1)==len(list2)
same_sum=sum(list1)==sum(list1)
common_values=any(num in list2 for num in list1)
print("Are the lists of the same length?",same_length)
print("Do the lists sum to the same values?",same_sum)
print("Is there any value that occurs in both lists?",common_values)
```

OUTPUT:

```
24mca35@projlabserver:~/pylab$ python3 work.py
Enter the first list of integers(comma seperated):10,20,30
Enter the second list of integers(comma seperated):40,50,90
Are the lists of the same length? True
Do the lists sum to the same values? True
Is there any value that occurs in both lists? False
```

PROGRAMME 13

AIM:
Program to count the occurence of each word in a line of text
SOURCE CODE:

```
text=input("Enter a line of text:")
words=text.split()
word_count={}
for word in words:
    word=word.lower()
    if word in word_count:
        word_count[word]+=1
    else:
        word_count[word]=1
print("word occurences:",word_count)
```

OUTPUT:

```
4mca35@projlabserver:~/pylab$ python3 lineof.py
nter a line of text:the sun rises in the east
ord occurences: {'the': 2, 'sun': 1, 'rises': 1, 'in': 1, 'east': 1}
```

PROGRAMME 14

AIM:
Program to show list comprehension
SOURCE CODE:

```
mbers=[-10,15,-3,7,-25,18,0]
positive_numbers=[num for num in numbers if num>0]
print(f"positive numbers in {numbers}:",positive_numbers)

N=5
squares=[num**2 for num in range (1,N+1)]
print("squares of first 5 numbers:",squares)

word="comprehension"
vowels=[char for char in word if char in 'aeiou']
print(f"vowels in the word:{word}",vowels)

word="hello"
ordinal_values=[ord(char) for char in word]
print("ordinal values of each character in the word:hello",ordinal_values)
```

OUTPUT:

```
24mca35@projlabserver:~/pylab$ python3 com.py
positive numbers in [-10, 15, -3, 7, -25, 18, 0]: [15, 7, 18]
squares of first 5 numbers: [1, 4, 9, 16, 25]
vowels in the word:comprehension ['o', 'e', 'e', 'i', 'o']
ordinal values of each character in the word:hello [104, 101, 108, 108, 111]
```

PROGRAMME 15

AIM:
Program to sort dictionaries
SOURCE CODE:

```
my_dict={'banana':3,'apple':5,'orange':2,'kiwi':4}
keys_asc=dict(sorted(my_dict.items()))
print("sorted by keys (ascending):",keys_asc)
keys_desc=dict(sorted(my_dict.items(),reverse=True))
print("Sorted by keys (descending):",keys_desc)

values_asc=dict(sorted(my_dict.items(),key=lambda item:item[1]))
print("Sorted by values (ascending):",values_asc)

values_desc=dict(sorted(my_dict.items(),key=lambda item:item[1],reverse=True))
print("Sorted by values (descending):",values_desc)
```

OUTPUT:

```
24mca35@projlabserver:~/pylab$ python3 show.py
sorted by keys (ascending): {'apple': 5, 'banana': 3, 'kiwi': 4, 'orange': 2}
Sorted by keys (descending): {'orange': 2, 'kiwi': 4, 'banana': 3, 'apple': 5}
Sorted by values (ascending): {'orange': 2, 'banana': 3, 'kiwi': 4, 'apple': 5}
Sorted by values (descending): {'apple': 5, 'kiwi': 4, 'banana': 3, 'orange': 2}
```

PROGRAMME 16

AIM:
Program to merge two dictionaries

SOURCE CODE:

```
dict1={'banana':3,'orange':5}
```

```
dict2={'apple':2,'kiwi':4}
print(dict1)
print(dict2)
dict1.update(dict2)
print(f"Merged:{dict1}")
```

OUTPUT: