**CYCLE 2**

**PROGRAM 1**

**Aim** : Program to create a string from the given string where the first and last characters are exchanged.

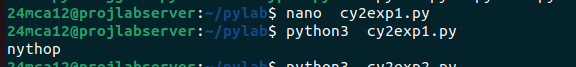
**Source code :**

str="python"

newstr=str[-1]+str[1:-1]+str[0]

print(newstr)

**Output :**

****

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**PROGRAM 2**

**Aim :** Program to get a string from an input string where all occurences of first character are replaced with ‘$’,except for first character.

**Source code :**

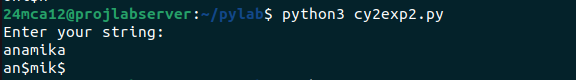
s=input("Enter your string:\n")

f=s[0]

newstr=f+s[1:].replace(f,'$')

print(newstr)

**Output :**

****

**PROGRAM 3**

**Aim :** Program to create a single string seprated with space from teo strings by swapping the characters position 1.

**Source code :**

string1=input("Enter first string:")

string2=input("Enter second string:")

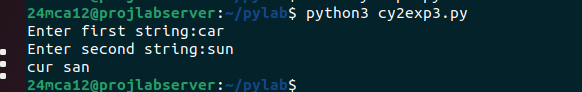
swap\_Str1=string1[0]+string2[1]+string1[2:]

swap\_Str2=string2[0]+string1[1]+string2[2:]

string3=swap\_Str1+" "+swap\_Str2

print(string3)

**Output :**

****

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**PROGRAM 4**

**Aim :** Program to counting the umber of characters in a string.

**Source code :**

n=input("Enter the string:").lower()

s={}

for i in n:

if i in s:

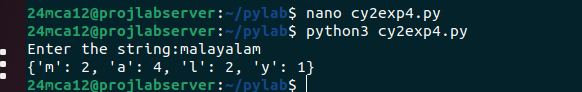
s[i]+=1

else:

s[i]=1

print(s)

**Output :**

****

**PROGRAM 5**

**Aim :** Program to add ‘ing’ at the end of a given string ,if it already ends with ‘ing’ add ‘ly’ .

**Source code :**

string=input("Enter a string:")

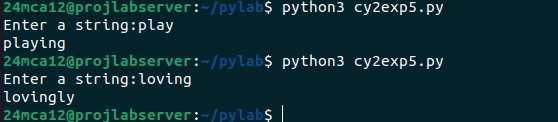
if "ing" in string:

print(string+"ly")

else:

print(string+"ing")

**Output :**

****

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PROGRAM 6**

**Aim :** Program to store a list of first names then count the occurance of ‘a’ within the list .

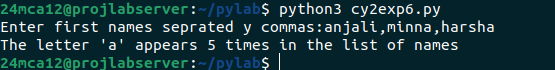
**Source code :**

names=input("Enter first names seprated y commas:")

count\_a=names.lower().count('a')

print(f"The letter 'a' appears {count\_a} times in the list of names")

**Output :**

****

**PROGRAM 7**

**Aim :** Program to read teo lists, prit out all colors from color\_list 1 not color\_list2.

**Source code :**

list1=input("Enter colors for list 1 seprated by comma:")

list2=input("Enter colors for list2 seprated by comma:")

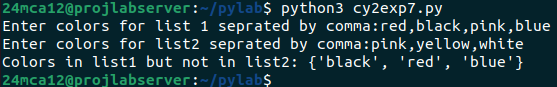
set1=set(list1.split(','))

set2=set(list2.split(','))

difference=set1-set2

print("Colors in list1 but not in list2:",difference)

**Output :**

****

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PROGRAM 8**

**Aim :** Program to create a list of colors and display first and last color .

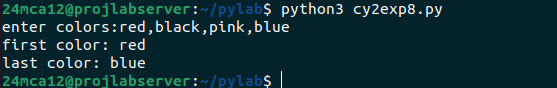
**Source code :**

colors=input("enter colors:").split(',')

print("first color:",colors[0])

print("last color:",colors[-1])

**Output :**

****

**PROGRAM 9**

**Aim :** Program to prompt the user for a list of integers ,for all values greater than 100 store ‘over’ instead.

**Source code :**

n=int(input("Enter number of integers to input:"))

list1=[]

for i in range(n):

num=int(input("Enter integers:"))

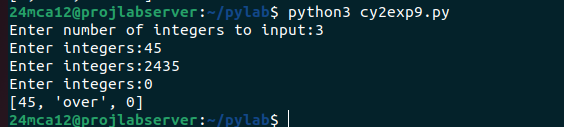
if num>100:

num="over"

list1.append(num)

print(list1)

**Output :**

****

**PROGRAM 10**

**Aim :** Program to form a list of integers ,create a list after removing even numbers.

**Source code :**

n=int(input("Enter number of integers to input:"))

listed=[]

for i in range(n):

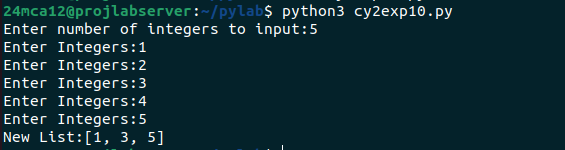
num=int(input("Enter Integers:"))

if num%2!=0:

listed.append(num)

print(f"New List:{listed}")

**Output :**



**PROGRAM 11**

**Aim :** Program to accept a list of words and return the length of longest word.

**Source code :**

str=input("Enter the list of words seprated by space:")

words=str.split()

length=0

for i in words:

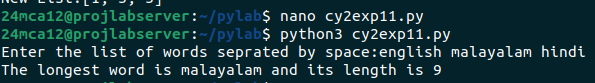
if len(i)>length:

longestword=i

length=len(i)

print(f"The longest word is {longestword} and its length is {length}")

**Output :**

****

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PROGRAM 12**

**Aim :** 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 :**

n=int(input("Enter  number of integers to input:"))

list1=[]

for i in range(n):

num=int(input("Enter integers:"))

list1.append(num)

n=int(input("Enter  number of integers to input:"))

list2=[]

for i in range(n):

num=int(input("Enter integers:"))

list2.append(num)

if len(list1)==len(list2):

print("lists are of the same length.")

else:

print("The lists are of different lengths.")

if sum(list1) == sum(list2):

print("The lists sum to the same value.")

else:

print("The lists do not sum to the same value.")

common\_value = set(list1).intersection(list2)

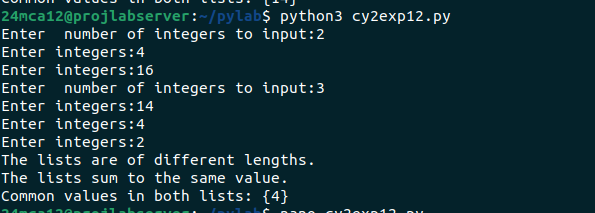
if common\_value:

print(f"Common values in both lists: {common\_value}")

else:

print("There are no common values in both lists.")

**Output :**

****

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PROGRAM 13**

**Aim :** Program to count the occurance 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:

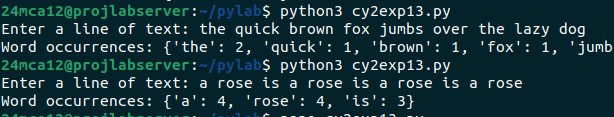
word\_count[word] += 1

else:

word\_count[word] = 1

print("Word occurrences:", word\_count)

**Output :**

****

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PROGRAM 14**

**Aim :** Program to show list comprehensions.

**Source code :**

numbers = [-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']

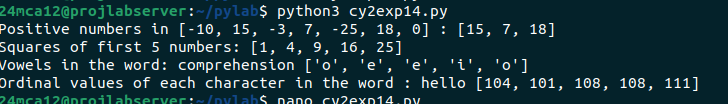
print(f"Vowels in the word: {word}", vowels)

word = "hello"

ordinal\_values = [ord(char) for char in word]

print("Ordinal values of each character in the word : hello", ordinal\_values)

**Output :**

****

**PROGRAM 15**

**Aim :** Program to sort dictionaries in ascending and discending order.

**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)

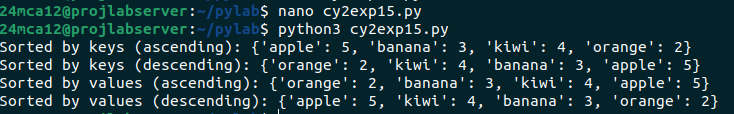
values\_asc = dict(sorted(my\_dict.items(), key=lambda item: item[1]))

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 :**

****

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PROGRAM 16**

**Aim :** Program to merge two dictionaries.

**Source code :**

dict1 = {'banana': 3, 'apple': 5}

dict2 = {'orange': 2, 'kiwi': 4}

print(dict1)

print(dict2)

dict1.update(dict2)

print(f"Merged :{dict1}")

**Output :**

