**PYTHON ASSINGMENT**

Name: ?????

Section: CSE – D

1. Write a program to check the given string is palindrome or not.

ALGORITHM

* Step 1: Start
* Step 2: Enter the String
* Step 3: Calulate rev\_str=str[::-1]
* Step 4: Check str = rev\_str , then print ‘The Given String is Palindrome’
* Step 5: Else print ‘The Given String is Not a Palindrome’
* Step 6: Stop

CODE

Graphical user interface, text, application, Word

Description automatically generatedGraphical user interface, text, application, Word

Description automatically generated

1. Write a program to print concatenation of first and last name of 10 students.

ALGORITHM

* Step 1: Start
* Step 2: Initialize fst and lst as List
* Step 3: For i in range (10)
* Step 4: Get input of first name and Second name of of first person
* Step 5: append the name in ele1 and ele2
* Step 6: For i in range (10) , print First name with last name
* Step 7: Stop

CODE

Graphical user interface, text, application

Description automatically generatedGraphical user interface, text, application

Description automatically generated

1. Write a program to slice the word ‘college’ from the sentence Sona College of technology.

ALGORITHM

* Step 1: Start
* Step 2: Initialize a = ‘Sona College Of Technology’
* Step 3: Stop

CODE

Graphical user interface, application, Word

Description automatically generatedGraphical user interface, application, Word

Description automatically generated

1. Write a python program to display all the common characters between two strings. Return -1 if there are no matching characters.

**Note**: Ignore blank spaces if there are any. Perform case sensitive string comparison wherever necessary.

|  |  |
| --- | --- |
| **Sample Input** | **Expected output** |
| "I like Python" "Java is a very popular language" | lieyon |

ALGORITHM

* Step 1: Start
* Step 2: Enter the First and Second String
* Step 3: a=(set(s1) & set(s2))
* Step 4: Check string length == 0
* Step 5: Else, print ‘The common letters’
* Step 6: Stop

CODE

**Graphical user interface, text, application, Word

Description automatically generatedGraphical user interface, text, application, Word

Description automatically generated**

1. Write a python program which accepts a string and removes all duplicate characters from the given string and return it.

|  |  |
| --- | --- |
| **Sample Input** | **Expected Output** |
| 1122334455ababzzz@@@123#\*#\* | 12345abz@#\* |

ALGORITHM

* Step 1: Start
* Step 2: Enter the String
* Step 3: Initialize ou = ‘ ‘
* Step 4: For string in str, check string not in ou
* Step 5: Then Calculate ou = ou+ string
* Step 6: Stop

CODE

Graphical user interface, application, Word

Description automatically generatedGraphical user interface, application, Word

Description automatically generated