1. Python Program to Check if a String is Palindrome or Not

def Palinmore(*s*):

    return *s*== *s*[::-1]

s=str(input("Nhap chuoi: "))

kq = Palinmore(s)

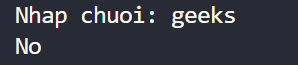
if kq:

    print("Yes")

else:

    print("No")





1. Python program to check whether the string is Symmetrical or Palindrome

string = str(input("Nhap chuoi: "))

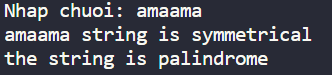
half = int(len(string) / 2)

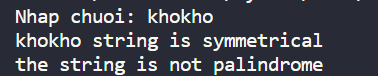
kytudau = string[:half]

kytu\_thuhai = string[half:]

print(string, 'string is symmetrical') if kytudau == kytu\_thuhai else print(string, 'string is not symmetrical')

print("the string is palindrome" if string==string[::-1] else "the string is not palindrome")





1. Reverse Words in a Given String in Python

def reverseWords(*s*):

    words = *s*.split(' ')

    reverse\_str = ' '.join(reversed(words))

    return reverse\_str

kq = str(input("Nhập chuỗi: "))

print ('Kết quả là: ',reverseWords(kq))



1. How to Remove Letters From a String in Python

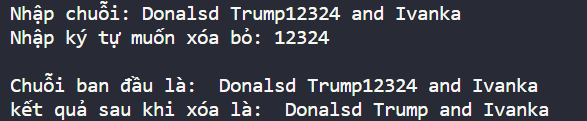
string = str(input("Nhập chuỗi: "))

remove = str(input("Nhập ký tự muốn xóa bỏ: "))

kq= string.replace(remove,'')

print("\nChuỗi ban đầu là: ", string)

print("kết quả sau khi xóa là: ", kq)

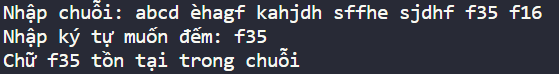


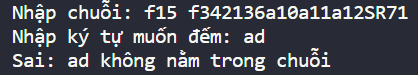
1. Check if String Contains Substring in Python

string = str(input("Nhập chuỗi: "))

dem = str(input("Nhập ký tự muốn đếm: "))

print(f"Chữ {dem} tồn tại trong chuỗi") if dem in string else print(f"Sai: {dem} không nằm trong chuỗi")





1. Python Words Frequency in String Shorthands

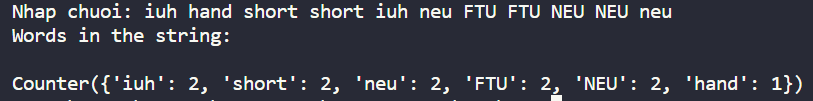
from collections import Counter

string = str(input("Nhap chuoi: "))

word= Counter(string.split())

print("Words in the string:\n")

print(word)



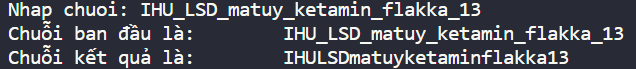
1. Convert Snake case to Pascal case

string = str(input("Nhap chuoi: "))

kq = string.replace('\_','')

print('Chuỗi ban đầu là: \t',string)

print('Chuỗi kết quả là: \t',kq)



1. Len of string in Python

string = str(input("Nhap chuoi: "))

def findLen(*str*):

    counter = 0

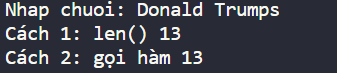
    for i in *str*:

        counter += 1

    return counter

print("Cách 1: len()", len(string))

print("Cách 2: gọi hàm", findLen(string))



1. Python program to print even length words in a string

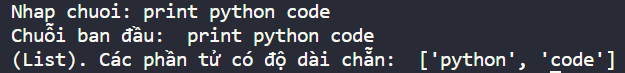
string = str(input("Nhap chuoi: "))

cach = string.split(' ')

kq =  list(filter(lambda *x*: len(*x*) %2 ==0, cach))

print("Chuỗi ban đầu: ", string)

print("(List). Các phần tử có độ dài chẵn: ", kq)



1. Python Program to Accept the Strings Which Contains all Vowels

string = str(input("Nhap chuoi: "))

def check(*string*):

    if len(set(*string*.lower()).intersection("aeiou")) >= 5:

        return ('accepted')

    else:

        return ("not accepted")

print(check(string))





1. Remove all duplicates from a given string in Python

def removeDuplicates(*s*, *n*):

    ans = ""

    for i in range (0,*n*):

        k=0

        for j in range (0,i):

            if(*s*[i]==*s*[j]):

                k=1

                break

        if(k==0):

            ans=ans+*s*[i]

    print(ans)

s = input("nhap chuoi: ")

removeDuplicates(s, len(s))



1. Python – Least Frequent Character in String

s = input("Nhap chuoi: ")

all ={}

for i in s:

    if i in all:

        all[i] += 1

    else:

        all[i] = 1

min = min(all)

print("Ký tự xuất hiện ít nhất trong chuỗi là: ",str(min))



1. Python –Max Frequent Character in String

s = input("Nhap chuoi: ")

all ={}

for i in s:

    if i in all:

        all[i] += 1

    else:

        all[i] = 1

max = max(all)

print("Ký tự xuất hiện nhiều nhất trong chuỗi là: ",str(max))

1. Program to check if a string contains any special character

import re

s = input("Nhap chuoi: ")

def Test(*str*):

    regex = re.compile('[@\_!#$%^&\*()<>?/\|}{~:]')

    print("String is accepted") if (regex.search(s) == None) else  print("String is not accepted.")

Test(s)





1. Python program to split and join a string

s = input("Nhap chuoi: ")

sp =s.split(' ')

print(sp)

print('-'.join(sp))

