**LISTA DE EXERCÍCIOS PYTHON COM STRINGS**

1. **Tamanho de strings.** Faça um programa que leia 2 strings e informe o conteúdo delas seguido do seu comprimento. Informe também se as duas strings possuem o mesmo comprimento e são iguais ou diferentes no conteúdo.

string\_1 = input(**"Digite a primeira string: "**)  
string\_2 = input(**"Digite a segunda string: "**)  
tamanho\_str\_1 = len(string\_1)  
tamanho\_str\_2 = len(string\_2)  
print(string\_1 + **" "** + str(tamanho\_str\_1))  
print(string\_2 + **" "** + str(tamanho\_str\_2))

1. **Nome ao contrário em maiúsculas.** Faça um programa que permita ao usuário digitar o seu nome e em seguida mostre o nome do usuário de trás para frente utilizando somente letras maiúsculas. Dica: lembre−se que ao informar o nome o usuário pode digitar letras maiúsculas ou minúsculas.

nome = input(**"Digite seu nome: "**)  
print(nome.upper() [::-1])

1. **Nome na vertical.** Faça um programa que solicite o nome do usuário e imprima-o na vertical.

nome = input(**'Digiete a palavra: '**)  
**for** i **in** nome:  
 print(str.upper(i))

1. **Nome na vertical em escada.** Modifique o programa anterior de forma a mostrar o nome em formato de escada.

nome = str(input(**'Digite seu nome: '**)).upper()  
**for** i **in** range(0,len(nome)+1):  
 print(nome[:i])

1. **Nome na vertical em escada invertida.** Altere o programa anterior de modo que a escada seja invertida.

nome = str(input(**'Digite seu nome: '**)).upper()  
**for** i **in** range(0,len(nome)+1):  
 print(nome[i:])

1. **Data por extenso.** Faça um programa que solicite a data de nascimento (dd/mm/aaaa) do usuário e imprima a data com o nome do mês por extenso.

dia, mes, ano = input(**'Data (dd/mm/aaaa): '**).split(**'/'**)  
meses = [**'janeiro'**, **'fevereiro'**, **'março'**, **'abril'**,  
 **'maio'**, **'junho'**, **'julho'**, **'agosto'**, **'setembro'**,  
 **'outubro'**, **'novembro'**, **'dezembro'**]  
print(**'Você nasce em: '**)  
print(**'%s de %s de %s'** %(dia, meses[int(mes)-1], ano))

1. **Conta espaços e vogais.** Dado uma string com uma frase informada pelo usuário (incluindo espaços em branco), conte:
   1. quantos espaços em branco existem na frase.
   2. quantas vezes aparecem as vogais a, e, i, o, u.

frase = input(**'Digite uma frase: '**)  
vogais = 0  
espacos = 0  
**for** letra **in** frase:  
 **if** letra == **" "**:  
 espacos += 1  
 **elif** letra **in "aeiou"**:  
 vogais += 1  
  
print(**"A frase tem %d vogais e %d espacos."** % (vogais,espacos))

1. **Palíndromo.** Um palíndromo é uma sequência de caracteres cuja leitura é idêntica se feita da direita para esquerda ou vice−versa. Por exemplo: **OSSO** e **OVO** são palíndromos. Em textos mais complexos os espaços e pontuação são ignorados. A frase **SUBI NO ONIBUS** é o exemplo de uma frase palíndroma onde os espaços foram ignorados. Faça um programa que leia uma sequência de caracteres, mostre−a e diga se é um palíndromo ou não.

expressao = input(**'Escreva uma expressao: '**).upper().replace(**' '**, **' '**)  
expInv = expressao [::-1]  
**if** expressao == expInv:  
 print(**'É palíndromo, pois, {} --> {}.'**.format(expressao, expInv))  
**else**:  
 print(**'Não é palíndromo.'**)

1. **Verificação de CPF.** Desenvolva um programa que solicite a digitação de um número de CPF no formato **xxx.xxx.xxx-xx** e indique se é um número válido ou inválido através da validação dos dígitos verificadores e dos caracteres de formatação.

**def** verificarNumeros(cpf):   
 **for** posicao,caractere **in** enumerate(cpf):  
 **if** posicao!=3 **and** posicao!=7 **and** posicao!=11 **and not** caractere.isdigit():  
 **return True  
 return False**cpf = input(**"CPF(xxx.xxx.xxx-xx) :"**) *#3 7 11***while** verificarNumeros(cpf) **or** cpf[3] !=**"." or** cpf[7] !=**"." or** cpf[11] !=**"-"**:  
 cpf = input(**"O 'CPF' pricisa estar no formato (xxx.xxx.xxx-xx) :"**)  
  
print(**"O 'CPF' está no formato correto"**)

1. **Número por extenso.** Escreva um programa que solicite ao usuário a digitação de um número até 99 e imprima-o na tela por extenso.

cont = (**'zero'**, **'um'**, **'dois'**, **'três'**, **'quatro'**,  
 **'cinco'**, **'seis'**, **'sete'**, **'oito'**, **'nove'**,  
 **'dez'**, **'onze'**, **'doze'**, **'treze'**, **'catorze'**,  
 **'quinze'**, **'dezesseis'**, **'dezessete'**, **'dezoito'**,  
 **'dezenove'**, **'vinte'**, **'vinte um'**, **'vinte e dois'**,  
 **'vinte e três'**, **'vinte e quatro'**, **'vinte e cinco'**,  
 **'vinte e seis'**, **'vinte e sete'**, **'vinte e oito'**,  
 **'vinte e nove'**, **'trinta'**, **'trinta e um'**, **'trinta e dois'**, **'trinta e três'**, **'trinta e quatro'**, **'trinta e cinco'**, **'trinta e seis'**, **'trinta e sete'**, **'trinta e oito'**, **'trinta e nove'**, **'quarenta'**, **'quarenta e um'**, **'quarenta e dois'**, **'quarenta e três'**, **'quarenta e quatro'**, **'quarenta e cinco'**, **'quarenta e seis'**, **'quarenta e sete'**, **'quarenta e oito'**, **'quarenta e nove'**,  
 **'cinquenta'**, **'cinquenta e um'**, **'cinquenta e dois'**, **cinquenta e três'**, **'cinquenta e quatro'**, **'cinquenta e cinco'**, **'cinquenta e seis'**, **'cinquenta e sete'**, **'cinquenta e oito'**,  
 **'cinquenta e nove'**, **'sessenta'**, **'sessenta e um'**, **'sessenta e dois'**, **'sessenta e três'**,  
 **'sessenta e quatro'**, **'sessenta e cinco'**, **'sessenta e seis'**, **'sessenta e sete'**, **'sessenta e oito'**,  
 **'sessenta e nove'**, **'setenta'**, **'setena e um'**, **'setenta e dois'**, **'setenta e três'**, **'setenta e quatro'**,  
 **'setenta e cinco'**, **'setenta e seis'**, **'setenta e sete'**, **'setenta e oito'**, **'setenta e nove'**,  
 **'oitenta'**, **'oitenta e um'**, **'oitenta e dois'**, **'oitenta e três'**, **'oitenta e quatro'**, **'oitenta e cinco'**,  
 **'oitenta e seis'**, **'oitenta e sete'**, **'oitenta e oito'**, **'oitenta e nove'**, **'noventa'**, **'noventa e um'**,  
 **'noventa e dois'**, **'noventa e três'**, **'noventa e quatro'**, **'noventa e cinco'**, **'noventa e seis'**,  
 **'noventa e sete'**, **'noventa e oito'**, **'noventa e nove'**)  
**while True**:  
 número = int(input(**"Digite um número entre 0 e 99: "**))  
 **if** 0<= número <= 99:  
 **break**

print(**'Tente novamente. '**, end=**''** )  
print(**f'Você digitou o número {**cont[número]**}'**)

1. **Jogo de Forca.** Desenvolva um jogo da forca. O programa terá uma lista de palavras lidas de um arquivo texto e escolherá uma aleatoriamente. O jogador poderá errar 6 vezes antes de ser enforcado.

**import** random  
palavras = input(**'Digite as palavras: '**)  
palavras = palavras.split(**" "**)  
  
uma\_palavra = palavras[random.randrange(0,len(palavras))]  
palavra\_forca = [**'\_'for** i **in** uma\_palavra]  
  
chance = 1  
**while**(chance < 7 **and** palavra\_forca.count(**"\_"**) !=0):  
 letra = input(**"Digite uma letra: "**)  
 **if** (letra **in** uma\_palavra):  
 print(**"A palavra é: "**, end=**""**)  
 **for** p **in** range (len(uma\_palavra)):  
 **if** letra == uma\_palavra [p]:  
 **del** palavra\_forca [p]  
 palavra\_forca.insert(p,letra)  
 print(**" "**.join(palavra\_forca))  
 **else**:  
 print(**"-> Você errou pela "** + str(chance) + **"a vez. Tente de novo!"**)  
 chance = chance + 1  
 **if** palavra\_forca.count(**"\_"**) == 0:  
 print(**"Parabéns! Você acertou a palavra."**)  
 **else**:  
 print(**"Forca! Fim de jogo."**)

1. **Valida e corrige número de telefone**. Faça um programa que leia um número de telefone, e corrija o número no caso deste conter somente 7 dígitos, acrescentando o '3' na frente. O usuário pode informar o número com ou sem o traço separador.

print( **'Valida e corrige número de telefone'** )  
numero = int ( input ( **'Telefone:'** ))  
novoNumero = **''  
if** len( str ( numero )) < 8 :  
 diferenca = 8 - len ( str ( numero ))  
 novoNumero = **'3'** \* diferenca  
  
numeroFormatado = novoNumero + str ( numero )  
numeroFormatado = numeroFormatado [ 0 : 4 ] + **'-'** + numeroFormatado [ 5 :]  
  
print ( **'Telefone possui% d dígitos. Vou adicionar o digito três na frente.'** % len ( str ( numero )))  
print ( **'Telefone corrigido sem formatação:% d'** % numero )  
print ( **'Telefone corrigido com formatação:% s'** % numeroFormatado )

1. **Jogo da palavra embaralhada.** Desenvolva um jogo em que o usuário tenha que adivinhar uma palavra que será mostrada com as letras embaralhadas. O programa terá uma lista de palavras lidas de um arquivo texto e escolherá uma aleatoriamente. O jogador terá seis tentativas para adivinhar a palavra. Ao final a palavra deve ser mostrada na tela, informando se o usuário ganhou ou perdeu o jogo.

**import** random  
  
tentativa = 7  
palavras = **'TESTE'**,**'PROGRAMA'**,**'SHOW'**,**'OVO'**,**'VIVAOLINUX'**,**'PYTHON'**,**'LUCAS'**,**'LINUX'**,**'LIMOSINE'**,**'FERRARI'**,**'CAMARO'**,**'BRANCO'**,**'PESQUISAR'**sorteado = random.choice(palavras)  
**while** tentativa != 0:  
 embaralha = random.sample(sorteado, len(sorteado))  
 juntar\_palavra\_embaralhada = **''**.join(embaralha)  
 print(juntar\_palavra\_embaralhada)  
 print(**"="**\*20)  
 tent = input(**"Digite a palavra: "**).upper()  
 **if** tent == sorteado:  
 print(**"\nParabens! Voce venceu!"**)  
 **break  
 else**:  
 tentativa -= 1  
 print(**"\nVoce errou! Tentativas restantes {tentativa}"**)  
 print(**"="**\*20)

1. **Leet spek generator.** Leet é uma forma de se escrever o alfabeto latino usando outros símbolos em lugar das letras, como números por exemplo. A própria palavra leet admite muitas variações, como l33t ou 1337. O uso do leet reflete uma subcultura relacionada ao mundo dos jogos de computador e internet, sendo muito usada para confundir os iniciantes e afirmar-se como parte de um grupo. Pesquise sobre as principais formas de traduzir as letras. Depois, faça um programa que peça uma texto e transforme-o para a grafia leet speak.

String = input(**'Insira o texto necessário: '**)  
  
**for** char **in** string:  
 **if** char == **'a'**:  
 string = string.replace(**'a'**,**'4'**)  
 **elif** char == **'b'**:  
 string = string.replace(**'b'**,**'8'**)  
 **elif** char == **'e'**:  
 string = string.replace(**'e'**,**'3'**)  
 **elif** char == **'l'**:  
 string = string.replace(**'l'**,**'1'**)  
 **elif** char == **'o'**:  
 string = string.replace(**'o'**,**'0'**)  
 **elif** char == **'s'**:  
 string = string.replace(**'s'**,**'5'**)  
 **elif** char == **'t'**:  
 string = string.replace(**'t'**,**'7'**)  
 **else**:  
 **pass**print(string)