

Acervo de Códigos Python: Soluções para Problemas Resolvidos

Uma Coleção Organizada de Códigos e Estratégias em Python

Introdução

A programação em Python consolidou-se como uma das ferramentas mais versáteis e populares tanto para iniciantes quanto para profissionais avançados em ciência da computação, engenharia, análise de dados e diversas áreas tecnológicas. Um dos melhores métodos para aprimorar habilidades em programação é resolver problemas práticos, seja por meio de competições, desafios online ou exercícios acadêmicos. Manter um acervo de códigos com base em problemas resolvidos torna-se, desse modo, um recurso valioso para consulta, revisão e aprendizado contínuo.

Este documento propõe a criação estruturada de um acervo de códigos Python, explorando como colecionar, organizar, documentar e ampliar soluções para problemas resolvidos, promovendo o crescimento técnico e colaborativo de estudantes, pesquisadores e profissionais da área.

Códigos

Abaixo estão os códigos para situações conceituais:

Quem é o maior e o menor?

Leia o peso de 5 pessoas e avalie quem possui o MAIOR e o MENOR peso.

```
maior = 0
menor = 0
for c in range(1,6):
    peso = float(input('Entre com um valor para {}º o peso: '.format(c)))
    if c == 1:
        maior = peso
        menor = peso
    else:
        if peso > maior:
            maior = peso
        if peso < menor:
            menor = peso
print('-'*20)
print('O maior peso lido foi de {}Kg'.format(maior))
print('O menor peso lido foi de {} Kg'.format(menor))
```

```
# Desafio_65: Crie um programa que leia vários números inteiros
# pelo teclado. No final da execução, mostre a média entre todos
# os valores e qual foi o maior e o menor valor lido. O programa
# deve perguntar ao usuário se ele quer ou não continuar a digitar
# valores.
op = 'S'
soma = cont = media = maior = menor = 0
while op in 'Ss':
    num = int(input('Digite um número inteiro: '))
    soma += num
    cont += 1
    if cont == 1:
        maior = menor = num
    else:
        if num > maior:
            maior = num
        if num < menor:
            menor = num
    op = str(input('Deseja continuar? [S/N]: ')).upper().strip()[0]
media = soma / cont
print('Você digitou {} números e a média foi {}'.format(cont,media))
print('O maior valor foi {} e o menor foi {}'.format(maior, menor))
```

```

# Crie um programa que peça para que o usuário continue informando
# números inteiros. O programa deve armazenar tais números em uma lista. O programa
# deve parar de capturar novos números caso o usuário insira 0 (zero).
# Ao final, o programa deve informar a quantidade de elementos adicionados na
# lista, bem como, o menor e o maior elemento digitado (excluindo o zero).
# By: RWT - 29/07/2025 - 17:26h
# Ref: https://www.youtube.com/watch?v=BudlaI2cwGk&list=PLpaKFn4Q4GMN1A4JJ1FnHW\_an0Gt8ug8ip&index=17
#
lista = []
num = 0
ma = 0
me = 0
soma = 0
cont = 0

while True:
    num = int(input("Digite um valor inteiro. Para sair[0]:"))
    soma += num
    cont += 1

    if num == 0:
        break
    if cont == 1:
        ma = me = num
    else:
        if num > ma:
            ma = num
        if num < me:
            me = num
    lista.append(num)

print(lista)
print(f"A lista possui {len(lista)} elementos.")
print(f"O maior número é {ma}")
print(f"O menor número é {me}")
print(f"A somatória dos elementos é {soma} e a média é {soma / (cont - 1):.2f}")

```

```

# Crie um programa que peça para o usuário preencher o conteúdo de uma matriz
# de dimensões 3x4. Após inseridos os dados, realize uma busca na matriz e informe
# quais são os valores das linhas e colunas (posição) do maior e do menor elemento
# de toda a matriz.

# Inicializa a matriz 3x4 vazia
matriz = []

# Preenchimento da matriz pelo usuário
print("Preencha a matriz 3x4:")
for i in range(3):
    linha = []
    for j in range(4):
        valor = int(input(f"Digite o valor para posição [{i}][{j}]: "))
        linha.append(valor)
    matriz.append(linha)

# Inicializa variáveis para maior e menor valor
maior = matriz[0][0]
menor = matriz[0][0]
pos_maior = (0, 0)
pos_menor = (0, 0)

# Busca pelos valores máximo e mínimo
for i in range(3):
    for j in range(4):
        if matriz[i][j] > maior:
            maior = matriz[i][j]
            pos_maior = (i, j)
        elif matriz[i][j] < menor:
            menor = matriz[i][j]
            pos_menor = (i, j)

# Exibe os resultados
print(f"\nMaior valor da matriz: {maior} na posição [linha {pos_maior[0]}][coluna {pos_maior[1]}]")
print(f"Menor valor da matriz: {menor} na posição [linha {pos_menor[0]}][coluna {pos_menor[1]}]")

```

```

1  # Crie um programa que peça para o usuário preencher o conteúdo de uma matriz
2  # de dimensões 3x4. Após inseridos os dados, realize uma busca na matriz e informe
3  # quais são os valores das linhas e colunas (posição) do maior e do menor elemento
4  # de toda a matriz.
5
6  # Inicializa a matriz 3x4 vazia
7  matriz = []
8
9  # Preenchimento da matriz pelo usuário
10 print("Preencha a matriz 3x4:")
11 for i in range(3):
12     linha = []
13     for j in range(4):
14         valor = int(input(f"Digite o valor para posição [{i}][{j}]: "))
15         linha.append(valor)
16     matriz.append(linha)
17
18     # Inicializa variáveis para maior e menor valor
19     maior = matriz[0][0]
20     menor = matriz[0][0]
21     pos_maior = (0, 0)
22     pos_menor = (0, 0)
23
24     # Busca pelos valores máximo e mínimo
25     for i in range(3):
26         for j in range(4):
27             if matriz[i][j] > maior:
28                 maior = matriz[i][j]
29                 pos_maior = (i, j)
30             elif matriz[i][j] < menor:
31                 menor = matriz[i][j]
32                 pos_menor = (i, j)
33
34     # Exibe os resultados
35     print(f"\nMaior valor da matriz: {maior} na posição [linha {pos_maior[0]}][coluna {pos_maior[1]}]")
36     print(f"Menor valor da matriz: {menor} na posição [linha {pos_menor[0]}][coluna {pos_menor[1]}]")

```

Validação de Dados

```
# ----- VALIDAÇÃO DE DADOS -----  
# Faça um programa que leia o sexo de uma pessoa,  
# mas só aceite os valores 'M' ou 'F'.  
# Caso esteja errado, peça a digitação novamente  
# até ter um valor correto.  
sexo = str(input('Digite o seu sexo [M/F]: ')).strip().upper()[0]  
while sexo not in 'MmFf':  
    sexo = str(input('Dados inválidos! Por favor digite novamente! [M/F]: ')).strip().upper()[0]  
print('Informação registrada com sucesso! Sexo: {}'.format(sexo))
```

Contador de operações

```
# Melhore o jogo do DESAFIO_028 onde o computador
# vai "pensar" em um número entre 0 e 10. Só que agora
# o jogador vai tentar adivinhar até acertar, mostrando
# no final quantos palpites foram necessários para vencer.
# Ref: https://www.youtube.com/watch?v=-Qk0IHJ1Chw&list=PLHz\_AreHm4dm6wY0IW20Nyg12TAjmMGT-&index=60
# ESSE DESAFIO FOI DESENVOLVIDO POR RWT EM 10/10/2023
import random
palpite = 0
comp = random.randint(0,10)
print('\033[32m * * * * \033[m\033[33mJOGO DE ADIVINHAR\033[m\033[32m * * * * \033[m')
num = 1
palpite = 0
while num != comp:
    num = int(input('{ } - Digite um número entre 0 e 10: '.format(palpite+1)))
    palpite += 1
    if num > comp:
        print('\033[35mMenos...\033[m')
    elif num == comp:
        print('\033[35mÉ isso aí!\033[m')
    else:
        print('\033[35mMais...\033[m')
print('\033[32m* \033[m'*18)
print('      Você tentou \033[1;36m{ }\033[m vezes.'.format(palpite))
print('\033[32m* \033[m'*18)
```

Operação com MENU.

```
# Crie um programa que leia dois valores e mostre um menu com os
# seguintes comandos: 1) Somar 2) Multiplicar 3) Maior
# 4) Novos números 5) Sair do programa
# Seu programa deverá realizar a operação em cada caso.
from time import sleep
print('\033[32m-\033[m'*15)
n1 = int(input('Digite o primeiro valor: '))
n2 = int(input('Digite o segundo valor: '))
opcao = 0
print('\033[32m-\033[m'*15)
while opcao != 5:

    print('\nSelecione a opção desejada:
[1] SOMAR
[2] MULTIPLICAR
[3] MAIOR
[4] NOVOS NÚMEROS
[5] SAIR')
    opcao = int(input('Sua opção: '))
    if opcao == 1:
        soma = n1 + n2
        print('A soma dos números {} e {} é {}'.format(n1, n2, soma))
    elif opcao == 2:
        produto = n1 * n2
        print('O resultado da multiplicação dos números {} e {} é {}'.format(
            n1, n2, produto))
    elif opcao == 3:
        if n1 > n2:
            print('O número {} é maior que {}'.format(n1, n2))
        else:
            print('O número {} é maior que {}'.format(n2, n1))
    elif opcao == 4:
        n1 = int(input('Digite o primeiro valor: '))
        n2 = int(input('Digite o segundo valor: '))
    elif opcao == 5:
        print('o programa está sendo finalizado.')
    else:
        print('Opção inválida!! Tente novamente!')
        sleep(3)
print('\033[33mFechando...\033[m')
```

Scape

```
# Desafio_64: Crie um programa que leia vários números inteiros pelo
# teclado. O programa só vai parar quando o usuário digitar o valor
# 999, que é a condição de parada. No final, mostre quantos números
# foram digitados e qual a soma entre eles (desconsiderando o flag).
# Resolvido: RWT
total = 0
cont = 0
num = 0
while num != 999:
    num = int(input('Digite um número inteiro: '))
    if num == 999:
        print('Você saiu do programa!')
        exit()
    else:
        total = total + num
        cont = cont + 1
        print('A soma dos números digitados é {} e o número de entradas é
{}'.format(total, cont))
print('você saiu do programa!')
```

```

-----
# Desafio_65: Crie um programa que leia vários números inteiros
# pelo teclado. No final da execução, mostre a média entre todos
# os valores e qual foi o maior e o menor valor lido. O programa
# deve perguntar ao usuário se ele quer ou não continuar a digitar
# valores.
op = 'S'
soma = cont = media = maior = menor = 0
while op in 'Ss':
    num = int(input('Digite um número inteiro: '))
    soma += num
    cont += 1
    if cont == 1:
        maior = menor = num
    else:
        if num > maior:
            maior = num
        if num < menor:
            menor = num
    op = str(input('Deseja continuar? [S/N]: ')).upper().strip()[0]
media = soma / cont
print('Você digitou {} números e a média foi {}'.format(cont,media))
print('O maior valor foi {} e o menor foi {}'.format(maior, menor))

```

```

# Faça um programa que mostre a tabuada de vários números,
# um de cada vez, para cada valor digitado pelo usuário.
# O programa será interrompido quando o número solicitado
# for negativo.
# RESOLVIDO POR RWT EM 17/10/2023

```

```

num = tab = 0
while True:
    num = int(input('Informe a tabuada que deseja ver: '))
    if num < 0:
        break
    for i in range(1,11):
        print(f'{i} x {num} = {i * num}')
print('Você saiu do programa!')

```

```

-----
# Desafio_68: Faça um programa que jogue PAR ou ÍMPAR com
# o computador. O jogo será interrompido quando o jogador
# perder, mostrando o total de vitórias consecutivas que
# ele conquistou no final do jogo.
# DESAFIO RESOLVIDO POR RWT EM 17/10/2023
import random
vitorias = 0
while True:
    comp = random.randint(0, 9)
    user = int(input('Diga um número: '))
    res_jogo = (comp + user) % 2
    op = str(input('Par ou Ímpar? [P/I]: '))
    if op in 'Pp':
        if res_jogo == 0:
            print(f'Você jogou {user} e o computador jogou {comp}. A soma deu {comp + user}')
            print('Deu PAR, você venceu!')
            vitorias = vitorias + 1
        else:
            print(f'Você jogou {user} e o computador jogou {comp}. A soma deu {comp + user}')
            print('Deu ÍMPAR, você perdeu!')

```



```

        break
    if op in 'Ii':
        if res_jogo == 1:
            print(f'Você jogou {user} e o computador jogou {comp}. A soma deu {comp + user}')
            print('Deu ÍMPAR, você venceu!')
            vitorias = vitorias + 1
        else:
            print(f'Você jogou {user} e o computador jogou {comp}. A soma deu {comp + user}')
            print('Deu PAR, você perdeu!')
            break
print(f'GAME OVER!! você venceu {vitorias} vezes')

```

```

# Desafio_69: Crie um programa que leia a idade de várias pessoas.
# A cada pessoa cadastrada, o programa deverá perguntar se o usuário
# quer ou não continuar. No final, mostre:
# A) Quantas pessoas tem mais de 18 anos.
# B) Quantos homens foram cadastrados.
# C) quantas mulheres tem menos de 20 anos.

```

```

tot18 = totalH = mulher20 = 0
homemvelho = mulhervelha = 0
while True:
    idade = int(input('Idade: '))
    sexo = ' '
    while sexo not in 'MF':
        sexo = str(input('Sexo: ')).upper().strip()[0]
    if idade >= 18:
        tot18 += 1
    if sexo == 'M':
        totalH += 1
        if idade > homemvelho:
            homemvelho = idade
    if sexo == 'F' and idade < 20:
        mulher20 += 1
    if sexo == 'F':
        if idade > mulhervelha:
            mulhervelha = idade
    resp = ' '
    resp = str(input('Quer continuar? [S/N]')).upper().strip()[0]
    if resp == 'N':
        break
print('-'*30)
print('          ESTATÍSTICAS')
print('-'*30)
print(f'Total de pessoas com mais de 18 anos: {tot18}')
print(f'O homem mais velho tem {homemvelho} anos')
print(f'A mulher mais velha tem {mulhervelha} anos')
print(f'Ao todo, temos {totalH} homens cadastrados')
print(f'Existem {mulher20} mulheres com menos de 20 anos')

```

```

# Crie um programa que leia o nome de vários produtos. O programa
# deverá perguntar se o usuário vai continuar. No final, mostre:
# A) Qual é o total gasto.
# B) Quantos produtos custam mais de R$ 1.000.
# C) Qual é o nome do produto mais barato.

total = prod_caro = prod1000 = prod_barato = cont = 0
ref_preco = 0
nome_prod_caro = ''
nome_prod_barato = ''
while True:
    produto = str(input('Produto: ')).strip()
    preco = float(input('Preço: R$'))
    cont += 1
    total += preco
    if preco > 1000:
        prod1000 += 1
    if cont == 1 or preco < prod_barato:
        prod_barato = preco
        nome_prod_barato = produto
    resp = ''
    while resp not in 'SN':
        resp = str(input('Continuar [S/N]: ')).upper().strip()[0]
    if resp == 'N':
        break
print('{:-^45}'.format(' FIM DO PROGRAMA '))
print('-'*45)
print('{: ^45}'.format(' ESTATÍSTICAS '))
print('-'*45)
print(f'Total da compra: R${total:.2f}')
print(f'{prod1000} Produtos acima de R$ 1.000,00')
#print(f'Produto {nome_prod_caro} é o mais caro da compra e custa: R${prod_caro:.2f}')
print(f'{nome_prod_barato} é o produto mais barato e custa R${prod_barato:.2f}')

```

```
# Crie um programa que simule o funcionamento de um caixa eletrônico.
# No início, pergunte ao usuário qual será o valor a ser sacado (número
# inteiro) e o programa vai informar quantas cédulas de cada valor serão
# entregues.
# OBS: Considere que o caixa possui cédulas de R$50, R$20, R$10 e R$1.
# Ref: https://www.youtube.com/watch?v=\_XGgwltYpYk
```

```
print('-'*30)
print('{:^30}'.format('BANCO R4'))
print('-'*30)
valor = int(input('Valor que deseja sacar: '))
total = valor
ced = 50
total_ced = 0
while True:
    if total >= ced:
        total = total - ced
        total_ced = total_ced + 1
    else:
        if total_ced > 0:
            print(f'Total de {total_ced} cédulas de R${ced}')
        if ced == 50:
            ced = 20
        elif ced == 20:
            ced = 10
        elif ced == 10:
            ced = 1
        total_ced = 0
        if total == 0:
            break
print('-'*30)
print('Volte sempre ao Banco R4')
```

```
# Crie um programa que simule o funcionamento de um caixa eletrônico.
# No início, pergunte ao usuário qual será o valor a ser sacado (número
# inteiro) e o programa vai informar quantas cédulas de cada valor serão
# entregues.
# OBS: Considere que o caixa possui cédulas de R$50, R$20, R$10 e R$1.
# ESTE CÓDIGO FOI DESENVOLVIDO POR RWT EM 19/10/2023
```

```
print('-'*30)
print('{:^30}'.format('BANCO R4'))
print('-'*30)
saque = int(input('Valor do saque: R$ '))
print('-'*30)
valor = saque
ced_50 = valor // 50
t20 = valor - (ced_50 * 50)
ced_20 = t20 // 20
t10 = t20 - (ced_20 * 20)
ced_10 = t10 // 10
t1 = t10 - (ced_10 * 10)
ced_1 = t1 // 1
if ced_50 != 0:
    print(f'{ced_50} cédulas de R$ 50,00')
if ced_20 != 0:
    print(f'{ced_20} cédulas de R$ 20,00')
if ced_10 != 0:
    print(f'{ced_10} cédulas de R$ 10,00')
if ced_1 != 0:
    print(f'{ced_1} cédulas de R$ 1,00')
print('-'*30)
print('Volte sempre - BANCO R4')
```

```

# Crie um programa que simule o funcionamento de um caixa eletrônico.
# No início, pergunte ao usuário qual será o valor a ser sacado (número
# inteiro) e o programa vai informar quantas cédulas de cada valor serão
# entregues.
# OBS: Considere que o caixa possui cédulas de R$50, R$20, R$10 e R$1.
# Ref: https://www.youtube.com/watch?v= XGgwltYpYk

```

```

print('-'*30)
print('{:^30}'.format('BANCO R4'))
print('-'*30)
saque = int(input('Digite o valor do saque: R$ '))
print('-'*30)
total = saque
cedula = 50
qtde_cedulas = 0
while True:
    if total >= cedula:
        total = total - cedula
        qtde_cedulas = qtde_cedulas + 1
    else:
        if qtde_cedulas > 0:
            print(f'{qtde_cedulas} cédulas de R${cedula:.2f}')
        if cedula == 100:
            cedula = 50
        if cedula == 50:
            cedula = 20
        elif cedula == 20:
            cedula = 10
        elif cedula == 10:
            cedula = 5
        elif cedula == 5:
            cedula = 1
        qtde_cedulas = 0
        if total == 0:
            break
print('-'*30)
print('='*30)
print('Volte sempre ao Banco R4')

```

```

-----
# Crie um programa que tenha uma Tupla totalmente preenchida com uma
# contagem por extenso, de zero a vinte. Seu programa deverá ler o
# número pelo teclado (entre 0 e 20) e mostrá-lo por extenso.
# RESOLVIDO POR RWT EM 19/10/2023.

```

```

ext = ('zero', 'um', 'dois', 'três', 'quatro', 'cinco', 'seis', 'sete', 'oito', 'nove', 'dez',
'onze', 'doze', 'treze', 'catorze', 'quinze', 'dezesseis', 'dezesete', 'dezoito', 'dezenove',
'vinte')
while True:
    num = int(input('Digite um número entre 0 e 20: '))
    if num < 0 or num > 20:
        num = int(input('Digite um número entre 0 e 20: '))
        break
    print(f'Você digitou o número \033[33m{ext[num]}\033[m')

```

```

# Crie uma Tupla preenchida com os 20 primeiros colocados da Tabela do
# Campeonato Brasileiro de Futebol, na ordem de colocação. Depois mostre:
# A) Apenas os 5 primeiros colocados.
# B) Os últimos 4 colocados.
# C) uma lista com os times em ordem alfabética.
# D) Em que posição na Tabela está o time da Cuiabá.

times = ('Botafogo', 'Bragantino', 'Flamengo', 'Grêmio', 'Palmeiras', 'Athletico-PR', 'Atlético-
MG', 'Fortaleza', 'Fluminense', 'Cuiabá', 'São
Paulo', 'Internacional', 'Corinthians', 'Bahia', 'Cruzeiro', 'Vasco', 'Santos', 'Goiás', 'Coritiba', 'Amé
rica-MG')
print('-'*30)
print('OS 5 PRIMEIROS COLOCADOS:')
#print('-'*30)
for cont in range(0, 5):
    print(f'{cont+1} - {times[cont]}')
print('-'*30)
print('OS 4 ÚLTIMOS COLOCADOS:')
#print('-'*30)
for cont in range(len(times) - 4, len(times)):
    print(f'{cont + 1} - {times[cont]}')
print('-'*30)
print('OS TIMES EM ORDEM ALFABÉTICA:')
classif = sorted(times)
for cont in classif:
    print(cont)
print('-'*30)
pesq = 0
clube = 'São Paulo'
for pesq in range(0,19):
    if times[pesq - 1] == clube:
        print(f'O time {clube} está na {pesq}ª posição do Campeonato Brasileiro.')
# SOLUÇÃO DIFERENTE PARA RESOLVER A LOCALIZAÇÃO DO TIME NA TABELA:
print(f'o time {clube} está na {times.index(clube) + 1}ª posição da tabela.')

```

```

-----
# Desafio_74: Crie um programa que vai gerar cinco números aleatórios
# e colocar em uma Tupla. Depois disso, mostre a listagem de números
# gerados e também indique o menor e o maior valor que estão na Tupla.
# RESOLVIDO POR RWT EM 22/10/2023.

```

```

import random
maior = menor = 0
n1 = random.randint(0,99)
n2 = random.randint(0,99)
n3 = random.randint(0,99)
n4 = random.randint(0,99)
n5 = random.randint(0,99)
num = (n1,n2,n3,n4,n5)
for cont in range(0, len(num)):
    if cont == 0:
        maior = menor = num[cont]
    else:
        if num[cont] > maior:
            maior = num[cont]
        if num[cont] < menor:
            menor = num[cont]
print(num)
print(f'Maior: {maior}\nMenor: {menor}')
-----

```

```
# Desafio_75: desenvolva um programa que leia quatro valores pelo teclado
# e guarde-os em uma tupla. No final, mostre:
# A) Quantas vezes apareceu o número 9.
# B) Em que posição foi digitado o primeiro valor 3.
# C) Quais foram os números pares.
```

```
cont = i = 0
n1 = int(input('Digite um número: '))
n2 = int(input('Digite um número: '))
n3 = int(input('Digite um número: '))
n4 = int(input('Digite um número: '))
num = (n1,n2,n3,n4)
print('-'*30)
print(f'Você digitou os seguintes valores: {num}')
if num.count(9) == 0:
    print('O número 9 não apareceu nenhuma vez!')
else:
    print(f'O número 9 apareceu {num.count(9)} vezes')
if 3 in num:
    print(f'O número 3 apareceu na {num.index(3)+1}ª posição')
else:
    print('O número 3 não foi digitado.')
print('Números pares:')
for n in num:
    if n % 2 == 0:
        print(n,end=' ')
```

```
# Desafio_76: crie um programa que tenha uma tupla única com
# nomes de produtos e seus respectivos preços, na sequência.
# No final, mostre uma listagem de preços, organizando os
# dados em forma tabular.
# Ref: https://www.youtube.com/watch?v=Qp2cXfCHK2I
```

```
listagem = ('Cerveja importada (330 ml)',12.20,
            'Cerveja nacional (0,5 litros)',6.20,
            'Garrafa de vinho qual. média',40.10,
            'Água (garrafa de 1,5 litros)',3.40,
            'Alface (1 unidade)',3.50,
            'Cebolas (1kg)',5.10,
            'Batatas (1 kg)',5.30,
            'Tomates (1 kg)',7.80,
            'Laranjas (1kg)',4.7,
            'Bananas (1kg)',5.70,
            'Maçãs (1 kg)',8.80,
            'Queijo fresco (1 kg)',44.00,
            'Uma dúzia de ovos',10.30,
            'Arroz (1 kg)',5.80,
            'Um quilo de pão (1 kg)',7.50,
            'Leite (1 litro)',5.30)
print('-'*40)
print(f'{"LISTAGEM DE PREÇOS":^40}')
print('-'*40)
for pos in range(0,len(listagem)):
    if pos % 2 == 0:
        print(f'{listagem[pos]:.<30}',end='')
    else:
        print(f'R${listagem[pos]:>7.2f}')
print('-'*40)
```

```
# Desafio_78: Faça um programa que leia 5 valores numéricos
# e guarde-os em uma lista. No final, mostre qual foi o maior
# e o menor valor digitado e as suas respectivas posições na lista.
# Ref: https://www.youtube.com/watch?v=q8Z1cRdJnfk
```

```
maior = menor = 0
valores = list()
for c in range(0,5):
    valores.append(int(input(f'Digite um valor para a posição {c}: ')))
    if c == 0:
        maior = menor = valores[c]
    else:
        if valores[c] > maior:
            maior = valores[c]

        if valores[c] < menor:
            menor = valores[c]
print('-'*50)
print(f'Você digitou os seguintes valores: {valores}')
print(f'O maior valor digitado foi {maior} nas posições ', end='')
for i, v in enumerate(valores):
    if v == maior:
        print(f'{i+1} - ', end='') # O incremento de 1 para o índice 'i' é opcional.
print()
print(f'O menor valor digitado foi {menor} nas posições ', end='')
for i, v in enumerate(valores):
    if v == menor:
        print(f'{i+1} - ', end='') # O incremento de 1 para o índice 'i' é opcional.
print()
print('-'*50)
```

```
# Desafio_79: Crie um programa onde o usuário possa digitar
# vários valores numéricos e cadastre-os em uma lista. Caso
# o número já exista lá dentro, ele não será adicionado. No
# final, serão exibidos todos os valores únicos digitados,
# em ordem crescente.
```

```
valores = []
while True:
    n = int(input('Digite um valor: '))
    if n not in valores:
        valores.append(n)
        print('Valor adicionado com sucesso!')
    else:
        print('Valor duplicado! - Não adicionado!')

    opcao = str(input('Deseja continuar [S/N]: '))
    if opcao in 'Nn':
        break
valores.sort()
print(f'Você digitou os seguintes valores: {valores}')
```



```

# Desafio_80: Crie um programa onde o usuário possa digitar cinco valores
# numéricos e cadastrá-los em uma lista, já na posição correta de inserção
# (sem usar sort()). No final, mostre a lista ordenada na tela.

lista = list()
ult_valor = 0
for i in range(0,5):
    n = int(input('Digite um valor: '))
    if i == 0 or n > lista[-1]: # list[len(lista)-1] Se for maior que o último valor.
        lista.append(n)
        print('Adicionado ao final da lista...')
    else:
        pos = 0
        while pos < len(lista):
            if n <= lista[pos]:
                lista.insert(pos,n)
                print(f'Adicionado na posição {pos} da lista...')
                break
            pos += 1
print('-=' * 30)
print(f'Os valores digitados em ordem foram {lista}')
-----

# Desafio_81: Crie um programa que vai ler vários números e colocar
# em uma lista. Depois disso, mostre:
# A) Quantos números foram digitados.
# B) A lista de valores, ordenada de forma decrescente.
# C) Se o valor 5 foi digitado e está ou não na lista.

cont = 0
lista = list()
while True:
    n = int(input('Digite um valor: '))
    cont += 1
    lista.append(n) # lista.append(int(input('Digite um valor: ')))
    opcao = str(input('Deseja continuar [S/N]: '))
    if opcao in 'Nn':
        break

lista.sort(reverse=True) # Ordem decrescente
print('-='*20)
print(f'Você digitou {cont} números') # Poderia ter usado: print(f' Você digitou {len(lista)
valores.}')
print(lista)
if 5 in lista:
    print('O número 5 faz parte da lista!')
else:
    print('O número 5 NÃO faz parte da lista!')
-----

```

```
# Desafio_82: Crie um programa que vai ler vários números e colocar em uma
# lista. Depois disso, crie duas listas extras que vão conter apenas os
# valores pares e os valores ímpares digitados, respectivamente. Ao final,
# mostre o conteúdo das três listas geradas.
```

```
# -----
#                                RESOLVIDO POR RWT EM 30/10/2023
# -----
```

```
valores = list()
pares = list()
impares = list()
while True:
    num = int(input('Digite um valor: '))
    valores.append(num)
    if num % 2 == 0:
        pares.append(num)
    else:
        impares.append(num)
    resp = str(input('Deseja continuar? [S/N]: '))
    if resp in 'Nn':
        break
print('-='*20)
print(f'Lista Geral: {valores}')
print(f'Valores pares: {pares}')
print(f'Valores ímpares: {impares}')
```

```
# Desafio_82: Crie um programa que vai ler vários números e colocar em uma
# lista. Depois disso, crie duas listas extras que vão conter apenas os
# valores pares e os valores ímpares digitados, respectivamente. Ao final,
# mostre o conteúdo das três listas geradas.
```

```
# Ref: https://www.youtube.com/watch?v=uk0gDFQEo\_I
```

```
num = list()
pares = list()
impares = list()
while True:
    num.append(int(input('Digite um número: ')))
    resp = str(input('Quer continuar? [S/N] '))
    if resp in 'Nn':
        break
for i, v in enumerate(num):
    if v % 2 == 0:
        pares.append(v)
    elif v % 2 == 1:
        impares.append(v)
print('-='*30)
num.sort()
print(f'A lista completa é {num}')
pares.sort()
print(f'A lista de pares é {pares}')
impares.sort()
print(f'A lista de ímpares é {impares}')
```

```
# Desafio_84: Faça um programa que leia nome e peso de várias pessoas,
# guardando tudo em uma lista. No final, mostre:
# A) Quantas pessoas foram cadastradas.
# B) Uma listagem com as pessoas mais pesadas.
# C) Uma listagem com as pessoas mais leves.
# Ref: https://www.youtube.com/watch?v=zPtvuLiEdKk
```

```
temp = list()
banco = list()
maior = menor = 0
while True:
    temp.append(str(input('Nome: ')))
    temp.append(int(input('Peso: ')))
    if len(banco) == 0:
        maior = menor = temp[1]
    else:
        if temp[1] > maior:
            maior = temp[1]
        if temp[1] < menor:
            menor = temp[1]
    banco.append(temp[:])
    temp.clear()
    resp = str(input('Deseja continuar? [S/N] '))
    if resp in 'Nn':
        break
print('-'*30)
# print(f'Os dados foram: {banco}')
print(f'Você cadastrou {len(banco)} pessoas')
print(f'O maior peso foi {maior}Kg. Peso de ', end='')
for p in banco:
    if p[1] == maior:
        print(f'[{p[0]}] ', end='')
print()
print(f'O menor peso foi {menor}Kg. Peso de ', end='')
for p in banco:
    if p[1] == menor:
        print(f'[{p[0]}] ', end='')
-----
```

```
# Desafio_85: Crie um programa onde o usuário possa digitar sete valores
# numéricos. Cadastre-os em uma lista única que mantenha separados os
# valores pares e ímpares. No final, mostre os valores pares e ímpares em
# ordem crescente.
# Ref: https://www.youtube.com/watch?v=2-fy24bbMJ4
```

```
valores = [[],[]]
valor = 0
for c in range(1,8):
    valor = int(input(f'Digite o {c}º valor da lista: '))
    if valor % 2 == 0:
        valores[0].append(valor)
    else:
        valores[1].append(valor)
print('-'*30)
valores[0].sort()
valores[1].sort()
print(f'Os valores pares digitados foram: {valores[0]}')
print(f'Os valores ímpares digitados foram: {valores[1]}')
-----
```

```
# Desafio_86: Crie um programa que crie uma matriz de dimensão 3x3 e
# preencha com valores lidos pelo teclado. No final, mostre a matriz
# na tela, com a formatação correta.
# Ref: https://www.youtube.com/watch?v=EGmlFdwD4C4
```

```
matriz = [[0,0,0],[0,0,0],[0,0,0]]
for l in range(0,3):
    for c in range(0,3):
        matriz[l][c] = int(input(f'Digite um valor para [{l},{c}]: '))
print('-'*30)
for l in range(0,3):
    for c in range(0,3):
        print(f'{matriz[l][c]:^5}', end='')
    print()
```

```
# Desafio_87: Aprimore o desafio anterior, mostrando no final:
# A) A soma de todos os valores pares digitados.
# B) A soma dos valores da terceira coluna.
# C) O maior valor da segunda linha.
# RESOLVIDO POR RWT EM 31/10/23
```

```
total = soma3col = 0
maior2lin = 0
matriz = [[0,0,0],[0,0,0],[0,0,0]]
for l in range(0,3):
    for c in range(0,3):
        matriz[l][c] = int(input(f'Digite um valor para [{l},{c}]: '))
        if matriz[l][c] % 2 == 0:
            total += matriz[l][c]
print('-'*30)
for l in range(0,3):
    for c in range(0,3):
        print(f'{matriz[l][c]:^5}', end='')
    print()
print('-'*30)
soma3col = matriz[0][2] + matriz[1][2] + matriz[2][2]

if matriz[1][0] > matriz[1][1]:
    if matriz[1][0] > matriz[1][2]:
        maior2lin = matriz[1][0]
    else:
        maior2lin = matriz[1][2]
elif matriz[1][1] > matriz[1][2]:
    maior2lin = matriz[1][1]
else:
    maior2lin = matriz[1][2]

print(f'A soma total de todos os elementos pares da matriz é {total}')
print(f'A soma da 3ª coluna da matriz é {soma3col}')
print(f'O maior valor da 2ª linha é {maior2lin}')
print('-'*30)
```

Desafio_88: Faça um programa que ajude um jogador da MEGA SENA a criar
palpites. O programa vai perguntar quantos jogos serão gerados e vai
sortear 6 números entre 1 e 60 para cada jogo, cadastrando tudo em uma
lista composta.
Ref: <https://www.youtube.com/watch?v=Hd7Ycaj61xE>

```
from random import randint
from time import sleep
lista = list()
jogos = list()
print('-'*30)
print('          JOGA NA MEGA SENA          ')
print('-'*30)
quant = int(input('Quantos jogos? '))
tot = 1
while tot <= quant:
    cont = 0
    while True:
        num = randint(1,60)
        if num not in lista:
            lista.append(num)
            cont += 1
        if cont >= 6:
            break
    lista.sort()
    jogos.append(lista[:])
    lista.clear()
    tot += 1
print('-=' * 3, f' SORTEANDO {quant} JOGOS ', '-=' * 3)
for i, l in enumerate(jogos): # i = índice, l = lista
    print(f'Jogo {i+1}: {l}')
    sleep(1)
print('-=' * 4, '< BOA SORTE! >', '-=' * 4)
```

Desafio_89: Crie um programa que leia o nome e duas notas de vários
alunos e guarde tudo em uma lista composta. No final, mostre um boletim
contendo a média de cada um e permita que o usuário possa mostrar as
notas de cada aluno individualmente.

Ref: <https://www.youtube.com/watch?v=7xrCJnniqMw>

```
ficha = list()
while True:
    nome = str(input('Nome: '))
    nota1 = float(input('Nota 1: '))
    nota2 = float(input('Nota 2: '))
    media = (nota1 + nota2) / 2
    ficha.append([nome, [nota1, nota2], media])
    resp = str(input('Quer continuar? [S/N] '))
    if resp in 'Nn':
        break
print('-='*30)
print(f'{"Nº":<4}{"NOME":<10}{"MÉDIA":>8}')
print('-'*26)
for i, a in enumerate(ficha):
    print(f'{i:<4}{a[0]:<8}{a[2]:>8.1f}') # Pode-se visualizar a numeração original: {i:<4}
while True:
    print('-'*35)
    opc = int(input('Mostrar notas de qual aluno? [999 interrompe]: '))
    if opc == 999:
        print('FINALIZANDO...')
        break
    if opc <= len(ficha) - 1:
        print(f'Notas de {ficha[opc][0]} são {ficha[opc][1]}')
print('<<< VOLTE SEMPRE >>>')
```

Desafio_90: Faça um programa que leia nome e média de um aluno, guardando também
situação em um dicionário. No final, mostre o conteúdo da estrutura na tela.
Ref: <https://www.youtube.com/watch?v=HipQYUk4koA&t=461s>

```
aluno = dict()
aluno['nome'] = str(input('Nome: '))
aluno['media'] = float(input(f'Média do aluno {aluno["nome"]}: '))
if aluno['media'] >= 7:
    aluno['situação'] = 'Aprovado'
elif 5 <= aluno['media'] < 7:
    aluno['situação'] = 'Recuperação'
else:
    aluno['situação'] = 'Reprovado'
print('-'*15)
for k, v in aluno.items():
    print(f' - {k} é igual a {v}')
```

Desafio-91: Crie um programa onde 4 jogadores joguem um dado e tenham
resultados aleatórios. Guarde esses resultados em um dicionário. No final,
coloque esse dicionário em ordem, sabendo que o vencedor tirou o maior
número no dado.
Ref: <https://www.youtube.com/watch?v=cwrqIztaAwk>

```
from random import randint
from time import sleep
from operator import itemgetter
jogo = {'Jogador-1': randint(1,6),
        'Jogador-2': randint(1,6),
        'Jogador-3': randint(1,6),
        'Jogador-4': randint(1,6)}
ranking = list()
print('Valores sorteados:')
for k, v in jogo.items():
    print(f'{k} tirou {v} no dado.')
    sleep(1)
ranking = sorted(jogo.items(), key=itemgetter(1), reverse=True)
print('-'*30)
print(' == RANKING DOS JOGADORES ==')
for i,v in enumerate(ranking):
    print(f' {i+1}º lugar: {v[0]} com {v[1]}.')
    sleep(1)
```

```
# Desafio_93: Crie um programa que gerencie o aproveitamento de um jogador
# de futebol. O programa vai ler o nome do jogador e quantas partidas ele
# jogou. Depois vai ler a quantidade de gols feitos em cada partida. No final,
# tudo isso será gravado em um dicionário, incluindo o total de gols feitos
# durante o campeonato.
# Ref: https://www.youtube.com/watch?v=5yKiud-YNnE
```

```
jogador = dict()
partidas = list()
jogador['nome'] = str(input('Nome do jogador: '))
tot = int(input(f'Quantas partidas {jogador["nome"]} jogou? '))
for c in range(0, tot):
    partidas.append(int(input(f'Quantos gols na partida {c + 1}: ')))
jogador['gols'] = partidas[:]
jogador['total'] = sum(partidas)
print('-'*30)
print(jogador)
print('-'*30)
for k, v in jogador.items():
    print(f'O campo {k} tem o valor {v}')
print('-'*30)
print(f'O jogador {jogador["nome"]} jogou {len(jogador["gols"])} partidas.')
for i, v in enumerate(jogador['gols']):
    print(f'    -> Na partida {i}, fez {v} gols.')
print(f'Foi um total de {jogador["total"]} gols.')
-----
```

```
# Desafio_94: Crie um programa que leia nome, sexo e idade de várias pessoas,  
# guardando os dados de cada pessoa em uma lista. No final, mostre:  
# A) Quantas pessoas foram cadastradas.  
# B) A média de idade do grupo.  
# C) Uma lista com todas as mulheres.  
# D) Uma lista com todas as pessoas com idade acima da média.  
# Ref: https://www.youtube.com/watch?v=ETnExBCFeps&t=13s
```

```
pessoa = dict() # INICIA A AQUISIÇÃO DE DADOS  
cadastro = list()  
soma_idade = media = 0  
while True:  
    pessoa.clear()  
    pessoa['nome'] = str(input('Nome: '))  
    while True:  
        pessoa['sexo'] = str(input('Sexo: [M/F] ')).upper()[0]  
        if pessoa['sexo'] in 'MF':  
            break  
        print('ERRO! Por favor, digite apenas M ou F.')  
    pessoa['idade'] = int(input('Idade: '))  
    soma_idade += pessoa['idade']  
    cadastro.append(pessoa.copy())  
    while True:  
        resp = str(input('Continuar? [S/N] ')).upper()[0]  
        if resp in 'SN':  
            break  
        print('ERRO! Responda apenas S ou N.')  
    if resp == 'N':  
        break  
print('-' * 30) # FIM DA AQUISIÇÃO DE DADOS  
# INICIA A ANÁLISE DE DADOS  
media = soma_idade / len(cadastro)  
print(f'A) Nº de pessoas cadastradas: {len(cadastro)}.')  
print(f'B) A média de idade do grupo: {media:5.2f}.')  
print('C) As mulheres cadastradas são: ')  
for p in cadastro:  
    if p['sexo'] in 'Ff':  
        print(f'{p["nome"]} ')  
print()  
print('D) Pessoas com idade acima da média de idade do grupo: ')  
for i in cadastro:  
    if i['idade'] >= media:  
        for k, v in i.items():  
            print(f' - {k} = {v}')  
print()  
-----
```



```
# Desafio_96: Faça um programa que tenha uma função chamada área(),
# que receba as dimensões de um terreno retangular (largura e
# comprimento) e mostre a área do terreno.
# Ref: https://www.youtube.com/watch?v=oV1s53YGtvE
```

```
def área(larg,comp):
    a = larg * comp
    print(f'A área de um terreno {larg}x{comp} é de {a}m².')
```

```
# PROGRAMA PRINCIPAL
print(' Controle de Terrenos')
print('-'* 20)
l = float(input('LARGURA (m): '))
c = float(input('COMPRIMENTO (m): '))
área(l,c)
```

```
-----
# Desafio_97: Faça um programa que tenha uma função chamada escreva(),
# que receba um texto qualquer como parâmetro e mostre uma mensagem
# com tamanho adaptável. Ex: escreva('Olá, Mundo!') Saída: Olá, Mundo!
# Ref: https://www.youtube.com/watch?v=Q6basnSo7r0
```

```
def escreva(msg):
    tam =len((msg)) + 4
    print('~' * tam)
    print(f'  {msg:^}')
    print('~' * tam)
```

```
# PROGRAMA PRINCIPAL
escreva('Ricardo Wagner Teixeira')
escreva('R4')
escreva('Curso de Python em vídeo no YouTube')
```