



COMPOSITION BOOK

100 Sheets 200 Pages
9 1/4 x 7 1/2 in / 24.7 x 19.0 cm

Exercício 072

Crie um programa que tenha uma tupla totalmente preenchida com uma contagem por extenso, de zero a vinte. Seu programa deverá ler um número pelo teclado (entre 0 e 20) e mostrá-lo por extenso.

```
cont = ('zero', 'Um', 'Dois', 'Tres', 'Quatro', 'Cinco', 'Seis', 'Sete',
        'Oito', 'Nove', 'Dez', 'Onze', 'Doze', 'Treze', 'Quatorze',
        'Quinze', 'Dezaseis', 'Dezasete', 'Dezoito', 'Dezanove', 'Vinte')
while True:
    num = int(input('Digite um número entre 0 e 20: '))
    if 0 <= num <= 20:
        break
    print('Tente novamente.', end="")
print(f'Você digitou o número {cont[num]}')
```

Exercicio 073

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 ordem alfabetica d) Em que posição está o time Chapecoense.

```
times = ('Corinthians', 'Palmeiras', 'Santos', 'Gremio', 'Cruzeiro',  
        'Flamengo', 'Vasco', 'Chapecoense', 'Atletico',  
        'Botafogo', 'Atletico-PR', 'Bahia', 'Sao Paulo',  
        'Fluminense', 'Sport Recife', 'EC Vitoria', 'Coritiba',  
        'Avai', 'Ponte Preta', 'Atletico-GO')
```

```
print('=-*15)  
print(f'Lista de times: {times}')  
print('=-*15)  
print(f'Os 5 primeiros sao: {times[0:5]}')  
print('=-*15)  
print(f'Os 4 últimos sao: {times[-4:]}')  
print('=-*15)  
print(f'Times em ordem alfabetica {sorted(times)}')  
print('=-*15)  
print(f'O chapecoense esta {times.index('Chapecoense')+1}º posição')
```

Exercício 074

Crie um programa que vai gerar cinco numeros aleatorios e colocar em uma tupla. Depois disso, mostre a listagem de numeros gerados e tambem indique o menor e o maior valor que estao na tupla.

```
from random import randint
numeros = (randint(1,10), randint(1,10), randint(1,10),
            randint(1,10), randint(1,10), )
print('Os valores sorteados foram: ', end="")
for n in numeros:
    print(f'{n}', end="")
print(f"\nO maior valor sorteado foi {max(numeros)}")
print(f'O menor valor sorteado foi {min(numeros)}')
```

✓ Ferramentas para Tuplas

Exercício 075

Desenvolva um programa que leia quatro valores pelo teclado e guarde-os em uma tupla. No final, mostre: a) Quantas vezes aparece o valor 9 b) Em que posição foi digitado o primeiro valor 3 c) Quais foram os números pares

```
num = (int(input('Digite um numero: '))  
       int(input('Digite outro numero: '))  
       int(input('Digite mais um numero: '))  
       int(input('Digite o ultimo numero: ')))  
print(f'Voce digitou os valores {num}')  
print(f'O valor 9 apareceu {num.count(9)}')  
if 3 in num:  
    print(f'O valor 3 apareceu na {num.index(3)+1}º posição')  
else:  
    print('O valor 3 não foi digitado em nenhuma posição')  
print('Os valores pares digitados foram: ', end="")  
for n in num:  
    if n % 2 == 0:  
        print(n, end="")
```

Exercício 076

Crie um programa que tenha uma tupla unica com nome e produtos e seus respetivos preços, na sequencia. No final, mostre uma listagem de preço, organizando os dados de forma tabular

```
listagem = ('Lapis', 1.75, 'Borracha', 2.00,  
           'Caderno', 15.90, 'Estojo', 25.00,  
           'Transferidor', 4.20, 'Compasso', 9.99,  
           'Mochila', 120.32, 'Canetas', 22.30)
```

```
print('*40)  
print(f'{“Listagem de Preço”:^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}’)  
print('*40)
```

Exercício 077

Crie um programa que tenha uma tupla com varias palavras (não usar acento). Depois disso, você deve mostrar para cada palavra, quais são as suas vogais.

```
palavras = ('aprender', 'programar', 'linguagem', 'python',  
            'curso', 'gratis', 'estudar', 'praticar', 'trabalhar',  
            'mercado', 'programador', 'futuro')
```

for p in palavras:

```
    print(f'\nNa palavra {p.upper()} temos: ', end="")
```

for letra in p:

```
    if letra.lower() in 'aeiou':
```

```
        print(letra, end="")
```

Se precisar de vogais com acento
é só colocá-las dentro do ''

Exercício 078

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.

```
listanum = []
mai = men = 0
for c in range(0,5):
    listanum.append(int(input(f'Digite um valor para a posição {c}: ')))
    if c == 0:
        mai = men = listanum[c]
    else:
        if listanum[c] > mai:
            mai = listanum[c]
        if listanum[c] < men:
            men = listanum[c]
print('*'*30)
print(f'Você digitou os valores {listanum}')
print(f'O maior valor digitado foi {mai} nas posições ', end="")
for i, v in enumerate(listanum):
    if v == mai:
        print(f'{i}...', end="")
print()
print(f'O menor valor digitado foi {men} nas posições ', end="")
for i, v in enumerate(listanum)
    if v == men:
        print(f'{i}...', end="")
```

Exercicio 079

Crie um programa onde o usuario possa digitar varios valores numericos e cadastre-os em uma lista. Caso o numero ja exista dentro, ele nao sera adicionado. No final, serao exibidos os valores unicos digitados, em ordem crescente.

```
numero = list()
while True:
    n = int(input('Digite um valor: '))
    if n not in numero:
        numero.append(n)
        print('Valor adicionado com sucesso...')
    else:
        print('Valor duplicado! Nao vou adicionar...')
    r = str(input('Quer continuar? [S/N] '))
    if r in 'Nn':
        break
print('*' * 30)
numero.sort()
print(f'Voce digitou os valores {numero}')
```

Se não existir o valor n
em numeros

Exercício 080

Crie um programa onde o usuário possa digitar 5 valores numéricos e cadastre-os em uma lista, já na posição correta de inserção, (sem usar o sort). No final, mostre a lista ordenada na tela.

```
lista = []
for c in range(0,5):
    n = int(input('Digite um valor: '))
    if c == 0 or n > lista[-1]: lista[-1] para descobrir o último elemento
        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 foram {lista}')
```

Exercicio 081

Crie um programa que vai ler varios numeros e colocar em uma lisa.

Depois disso, mostre:

- a) quantos numeros foram digitados
- b) a lista de valores, ordenada de forma decrescente
- c) se o valor 5 foi digitado e esta na lista ou nao

```
valores = []
```

```
while True:
```

```
    valores.append(int(input('Digite um valor: ')))
```

```
    resp = str(input('Quer continuar? [S/N] '))
```

```
    if resp in 'Nn':
```

```
        break
```

```
print('=-*30')
```

```
print(f'Voce digitou {len(valores)} elementos')
```

```
valores.sort(reverse=True)
```

```
print(f'Os valores em ordem decrescente sao {valores}')
```

```
if 5 in valores:
```

```
    print('O valor 5 faz parte da lista')
```

```
else:
```

```
    print('O valor 5 nao foi encontrado na lista')
```

Exercicio 082

Crie um programa que vai ler varios numeros e colocar em uma lisa.

Depois disso crie duas listas extras, que vao conter apenas valores pares e os valores impares respetivamente. No final, mostre o conteudo das 3 listas geradas.

```
pares = list()
impares = list()
num = list()
while True:
    num.append(int(input('Digite um numero: ')))
    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)
print(f'A lista completa é {num}')
print(f'A lista de pares é {pares}')
print(f'A lista de impares é {impares}')
```

Exercicio 083

Crie um programa onde o usuario digite uma expressao qualquer que use parenteses. Seu aplicativo devera analisar se a expressao passada esta com os parenteses abertos e fechados respectivamente.

```
expr = str(input('Digite a expressao: '))
pilha = []
for simb in expr:
    if simb == '(':
        pilha.append('(')
    elif simb == ')':
        if len(pilha) > 0:
            pilha.pop()
        else:
            pilha.append(')')
            break
    if len(pilha) == 0:
        print('Sua expressao esta valida')
    else:
        print('Sua expressao esta invalida')
```

Exercício 084

Faça um programa que leia o nome e o peso, de varias pessoas, guardando tudo em uma lista. No final, mostre: a) Quantas pessoas foram registradas b) Uma listagem com as pessoas mais pesadas c) Uma listaem com as pessoas mais leves

```
temp = []
princ = []
mai = men = 0
while True:
    temp.append(str(input('Nome: ')))
    temp.append(float(input('Peso: ')))
    if len(princ) == 0:
        mai = men = temp[0]
    else:
        if temp[0] > mai:
            mai = temp[0]
        if temp[0] < men:
            men = temp[0]
    princ.append(temp[:])
    temp.clear()
    resp = str(input('Quer continuar? [S/N] '))
    if resp in 'Nn':
        break
print('=-*30')
print(f'Ao todo, voce cadastrou {len(princ)} pessoas.')
print(f'O maior peso foi de {mai}kg. Peso de ', end="")
for p in princ:
    if p[1] == mai:
        print(f'{p[0]}', end=" ")
print()
print(f'O menor peso foi {men}Kg. Peso de ', end="")
for p in princ:
    if p[1] == men:
        print(f'{p[0]}', end=" ")
```

Exercicio 085

Crie um programa onde o usuario digite possa digitar sete valores numericos e cadastre-os em uma lista unica que mantenha separados os valores pares e impares. No final, mostre os valores pares e impares em ordem crescente.

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

Exercício 086

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.

```
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()
```

Exercício 087

Aprimore o desafio anterior, mostrando no final: a) A soma de todos os numeros pares digitados b) A soma dos valores da terceira coluna c) O maior valor da segunda linha

```
matriz = [[0,0,0],[0,0,0],[0,0,0]]
```

```
spar = mai = scol = 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='')
```

```
    if matriz[l][c] % 2 == 0:
```

```
        spar += matriz[l][c]
```

```
    print()
```

```
print(f'A soma dos valores pares é {spar}')
```

```
for l in range(0,3):
```

```
    if c == 0 or matriz[l][c] > mai:
```

```
        mai = matriz[l][c]
```

```
print(f'O maior valor da segunda linha é {mai}')
```

Exercício 088

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.

```
from random import randint
from time import sleep
lista = list()
jogos = list()
print('=-*30')
print('{:^30}'.format('JOGA NA MEGA SENA'))
print('=-*30')
quant = int(input('Quantos jogos você quer que eu sorteie? '))
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):
    print(f'Jogo {i+1}: {l}')
    sleep(1)
print(f'=-*5, < BOA SORTE >, '=-*5)
```

Exercício 089

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

```
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}{{"Media":>8}}}')
print('=-*26)
for i, a in enumerate(ficha):
    print(f'{i:<4}{{a[0]:<10}{a[2]:>8.1f}}')
while True:
    print('=-*30)
    opc = int(input('Mostrar notas de qual aluno? (999 interrompe) '))
    if opc == 999:
        print('Finalizando...')
        break
    if opc <= len(ficha):
        print(f'Notas de {ficha[opc][0]} sao {ficha[opc]}')
print('=<<< VOLTE SEMPRE >>>')
```

Exercicio 090

Faça um programa que leia nome e média de um aluno, guardando também a situação em um dicionário. No final, mostre o conteúdo da estrutura na tela.

```
aluno = dict()
aluno['nome'] = str(input('Nome: '))
aluno['media'] = float(input(f'Media de {aluno["nome"]}: '))
if aluno['media'] >= 7:
    aluno['situação'] = 'Aprovado'
elif 5 <= aluno['media']:
    aluno['situação'] = 'Recuperação'
else:
    aluno['situação'] = 'Reprovado'
print(f'=*30')
for k, v in aluno.items():
    print(f' - {k} é igual a {v}')
```

Exercicio 091

Crie um programa onde 4 jogadores joguem um dado e tenham resultados aleatorios. Guarde esses resultados em um dicionario. No final, coloque esse dicionario em ordem, sabendo que o vencedor tirou o maior numero.

```
from random import randint
from time import sleep
from operator import itemgetter
jogo = {'jogador1': randint(1,6), 'jogador2': randint(1,6), 'jogador3': randint(1,6),
        'jogador4': 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)
```

Operador para dicionários

Se for itemgetter(0) ele pega a chave, senão ele pega o valor

Exercicio 092

Crie um programa que leia nome, ano de nascimento e carteira de trabalho e cadastre-os (com idade) em um dicionario se por acaso o CTPS for diferente de 0, o dicionario recebera tambem o ano de contratação e o salario. Calcule e acrescente, além da idade, com quantos anos a pessoa vai se aposentar.

```
from datetime import datetime
dados = dict()
dados['nome'] = str(input('Nome: '))
nasc = int(input('Ano de nascimento: '))
dados['idade'] = datetime.now().year - nasc
dados['ctps'] = int(input('Carteira de trabalho (0 não tem): '))
if dados['ctps'] != 0:
    dados['contratação'] = int(input('Ano de contratação: '))
    dados['salario'] = float(input('Salário: R$'))
    dados['aposentadoria'] = dados['idade'] + ((dados['contratação'] + 35) -
    datetime.now().year)
for k, v in dados.items():
    print(f' - {k} tem o valor {v}')
```

Exercício 093

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 sera guardado em um dicionario, incluindo o total de gols feitos durante o campeonato.

```
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}: ')))
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 valot {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.')
```

Somar o que está dentro da lista 'partidas'

Exercicio 094

Crie um programa que leia nome, sexo e idade de varias pessoas, guardando os dados de cada pessoa em um dicionario e todos os dicionarios 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 media.

```
galera = list()
pessoa = dict()
soma = 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 += pessoa['idade']
    galera.append(pessoa.copy())
    while True:
        resp = str(input('Quer continuar? [S/N]')).upper()[0]
        if resp in 'SN':
            break
        if resp == 'N':
            break
    print('=-*30')
print(f'A) Ao todo temos {len(galera)} pessoas cadastradas.')
media = soma/ len(galera)
print(f'B) A média de idade é de {media:5.2f} anos')
print('C) As mulheres cadastradas foram', end=' ')
for p in galera:
    if p['sexo'] in 'F':
        print(f'{p["nome"]}', end=' ')
print()
print(f'D) Lista das pessoas que estão acima da média: ', end=' ')
for p in galera:
    if p['idade'] >= media:
        print(' ', end=' ')
        for k, v in p.items():
            print(f'{k} = {v}; ', end=' ')
        print()
print('=> ENCERRADO >>')
```

Exercício 095

Aprimore o desafio 093 para que ele funcione com varios jogadores, incluindo um sistema de visualização de detalhes do aproveitamento de cada jogador.

```
jogador = dict()
partidas = list()
time = list()
while True:
    jogador.clear()
    jogador['nome'] = str(input('Nome do jogador: '))
    tot = int(input(f'Quantas partidas {jogador["nome"]} jogou? '))
    partidas.clear()
    for c in range(0,tot):
        partidas.append(int(input(f' Quantos gols na parida {c+1}: ')))
    jogador['gols'] = partidas[:]
    jogador['total'] = sum(partidas)
    time.append(jogador.copy())
    while True:
        resp = str(input('Quer continuar? [S/N]')).upper()[0]
        if resp in 'SN':
            break
        if resp == 'N':
            break
    print(f'{"="*30}')
    print('cod', end=' ')
    for i in jogador.keys():
        print(f'{i}<15', end=' ')
    print()
    print(f'{"="*40}')
    for k, v in enumerate(time):
        print(f'{k}>3', end=' ')
        for d in v.values():
            print(f'{str(d)}<15', end=' ')
        print()
    print(f'{"="*40}')
    while True:
        busca == int(input('Mostrar dados de qual jogador? (999 para parar) '))
        if busca == 999:
            break
        else:
            for i, g in enumerate(time[busca]['gols']):
                print(f' No jogo{i+1} fez {g} gols')
```

Exercício 096

Faça um programa que tenha uma função chamada `area()`, que receba as dimensões de um terreno retangular e mostre a área do terreno.

```
def area(larg, comp):  
    a = larg * comp  
    print(f'A área de um terreno {larg}x{comp} é de {a}')  
  
print('Controle de Terrenos')  
print('-'*20)  
l = float(input('Largura (m): '))  
c = float(input('Comprimento (m): '))  
area(l,c)
```

Exercicio 097

Faça um programa que tenha uma função chamada `escreva()`, que receba um texto qualquer como parâmetro e mostre uma mensagem com tamanho adaptável.

```
def escreva(msg):  
    tam = len(msg) + 4  
    print('-'* tam)  
    print(f" {msg} ")  
    print('-'* tam)  
  
escreva(Mensagem)
```

Exercício 098

Faça um programa que tenha uma função chamada `contador()`, que receba 3 parâmetros: `inicio`, `fim` e `passo` e realize a contagem. Seu programa tem que realizar 3 contagens através da função criada. A) de 1 a 10, de 1 em 1 b) de 10 a 0, de 2 em 2 c) Uma contagem personalizada.

```
from time import sleep
def contador(i, f, p):
    print('-*30')
    if p < 0:
        p *= -1
    if p == 0:
        p = 1
    print(f'Contagem de {i} até {f} de {p} em {p}')
    sleep(2)
    if i < f:
        cont = i
        while cont <= f:
            print(f'{cont}', end=',', flush=True)
            sleep(0.5)
        print('FIM')
    else:
        cont = i
        while cont >= f:
            print(f'{cont}', end=',', flush=True)
            sleep(0.5)
        cont -= p
        print('FIM')

contador(1,10,1)
contador(10,0,2)
print('Agora é a sua vez de personalizar a contagem')
ini = int(input('Início: '))
fim = int(input('Fim: '))
passo = int(input('Passo: '))
contador(ini,fim,passo)
```