



Aula 18 - Variáveis compostas (Lista II)

▼ Dificuldade	★
☰ Tags	cópia for lista dentro de lista relação de listas

Em uma lista, é possível colocar **listas dentro delas**.

Se eu criar uma lista como dados, e dentro dela o nome de uma pessoa e a idade, criamos **uma** lista.

```
dados = []
dados.append('Pedro')
dados.append(25)
```



Adicionando uma lista dentro de outra

Agora criaremos uma **lista chamada pessoas** e ao invés de adicionarmos valores a ela, **adicionaremos a lista dados**.

```
dados = []
dados.append('Pedro')
dados.append(25)
pessoas.append(dados[:])
dados.clear()
```

A lista dados ficará no **índice 0** da lista pessoas.



A partir daí, podemos adicionar quantas listas quisermos, no caso, outras pessoas.



Podemos fazer isso manualmente, já colocando cada pessoa em uma **lista própria**.

```
pessoas = [['Pedro', 25], ['Maria', 19], ['João', 32]]
```

Mostrar a lista que está dentro de outra

Agora, podemos querer dar print **somente em Pedro**, para isso, temos que fatiar primeiro o índice que a lista em que Pedro está, depois de dentro dela, pegar o índice de Pedro.

```
print(pessoas[0][0])
```

Pedro será mostrado.

```
print(pessoas[2][1])
```

32 será mostrado.

```
print([1])
```

Será mostrado a lista inteira de Maria, ['Maria', 19].

Relação entre listas

Se eu colocar uma lista dentro da outra sem o uso do fatiamento geral [:], será criado uma **relação** entre elas.

```
test = []
test.append('Ian')
test.append(17)
galera = []
galera.append(test)
test[0] = 'Mário'
test[1] = 29
galera.append(test)
print(galera)
```

O print mostrará [['Mário', 29], ['Mário', 29]].

Agora se realmente **criarmos uma cópia**, teremos:

```
test = []
test.append('Ian')
test.append(17)
galera = []
galera.append(test[:])
test[0] = 'Mário'
test[1] = 29
galera.append(test)
print(galera)
```

['Ian', 17], ['Mário', 29]]

Estrutura for

Podemos criar uma **estrutura for** que passará por cada elemento da lista.

```
galera = [['Mario', 19], ['Maria', 21], ['Joaquim', 29]]
for c in galera:
    print(c)
```

Se quisermos **apenas** o **nome** e a **idade** de cada lista, basta fatiar com 0 e 1, respectivamente.

```
galera = [['Mario', 19], ['Maria', 21], ['Joaquim', 29]]
for c in galera:
```

```
print(f'{c[0]} tem {c[1]} anos.')
```

Criaremos um programa que receba **nome** e **idade** e os coloque em uma lista, depois, colocando ele em outra lista, verificamos quem é maior de idade e colocamos o nome da pessoa.

```
galera = []
dados = []
for c in range(0, 3):
    dados.append(input('Nome: '))
    dados.append(int(input('Idade: ')))
    galera.append(dados[:])

for p in galera:
    if p[1] >= 18:
        print(f'{p[0]} é maior de idade!')
```

Desafio 84

Faça um programa que leia nome e peso de várias pessoas, guardando tudo em uma lista. No final, mostre:

1. Quantas pessoas foram cadastradas.
2. Uma listagem com as pessoas mais pesadas.
3. Uma listagem com as pessoas mais leves.

```
personas = []
dados = []
while True:
    dados.append(input('Digite seu nome: '))
    dados.append(float(input('Digite seu peso: ')))
    personas.append(dados[:])
    dados.clear()
    r = input('Quer continuar? [S/N]: ')
    if r in 'Nn':
        break
print(f'{len(personas)} personas foram cadastradas!')
pesado = 0
leve = 0
nome_pesado = nome_leve = ''
for c in personas:
    if pesado == 0 and leve == 0:
        leve = c[1]
        pesado = c[1]
        nome_pesado = c[0]
        nome_leve = c[0]
    elif c[1] >= pesado:
        pesado = c[1]
```

```

        nome_pesado = c[0]
    elif c[1] <= leve:
        leve = c[1]
        nome_leve = c[0]
print(f'O maior peso foi de {pesado}kg. Peso de ', end='')
for cont in pessoas:
    if cont[1] == pesado:
        print(f'[{cont[0]}] ', end='')
print(f'\nO menor peso foi de {leve}kg. Peso de ', end='')
for p in pessoas:
    if p[1] == leve:
        print(f'[{p[0]}] ', end='')

```

<https://youtu.be/zPtvuLiEdKk>

Desafio 85

Crie um programa onde o usuário possa digitar sete valores numéricos e 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.

```

numeros = [[], []]
for c in range(0, 7):
    n = (int(input('Digite um número: ')))
    if n % 2 == 0:
        numeros[0].append(n)
    else:
        numeros[1].append(n)

numeros[0].sort()
print(f'Os números pares são: {numeros[0]}')
numeros[1].sort()
print(f'Os números ímpares são: {numeros[1]}')

```

<https://youtu.be/2-fy24bbMJ4>

Desafio 86

Crie um programa que crie uma matriz de dimensão 3×3 e preencha com valores lidos pelo teclado.

0			
1			
2			
	0	1	

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 o valor para a linha {l} e coluna {c}: '))

for l in range(0, 3):
    for c in range(0, 3):
        print(f'{matriz[l][c]^5} ', end='')
    print()
```

<https://youtu.be/EGmIFdwD4C4>

Desafio 87

Aprimore o desafio anterior, mostrando no final:

1. A soma de todos os valores pares digitados.
2. A soma dos valores da terceira coluna.
3. O maior valor da segunda linha.

```
matriz = [[0, 0, 0], [0, 0, 0], [0, 0, 0]]
par = soma3 = 0

for l in range(0, 3):
    for c in range(0, 3):
        n = int(input(f'Digite o valor para a linha {l+1} e coluna {c+1}: '))
        matriz[l][c] = n
        if n % 2 == 0:
            par += n

for l in range(0, 3):
    for c in range(0, 3):
        print(f'{matriz[c][l]^5} ', end='')
    print()
```

```
print()

print(f'A soma de todos os valores pares é: {par}')
```

```
print(f'A soma dos valores da coluna 3 é: {sum(matriz[c])} ')
print(f'O maior valor da segunda linha é: {max(matriz[1])}')
```

<https://youtu.be/QhS829x6up4>

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.

```
from random import randint
lista = []
jogo = []
n = int(input('Quantos jogos você quer sortear? '))
for c in range(0, n):
    for cont in range(0, 6):
        num = randint(1, 60)
        while True:
            while num in jogo:
                num = randint(1, 60)
            if num not in jogo:
                jogo.append(num)
                break
        lista.append(jogo[:])
        lista[c].sort()
    print(f'Jogo {c+1}: {lista[c]}')
    jogo.clear()
```

<https://youtu.be/Hd7Ycaj61xE>

Desafio 89

Crie um programa que leia 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 alunos individualmente.

```

dados = []
lista = []
n = 0
while True:
    dados.append(input('Nome: '))
    dados.append(float(input('Nota 1: ')))
    dados.append(float(input('Nota 2: ')))
    lista.append(dados[:])

    dados.clear()
    resp = input('Quer continuar? [S/N]: ')
    if resp in 'Nn':
        break
print('-' * 30)
print('No      Nome                MÉDIA')
print('-' * 30)
for n, p in enumerate(lista):
    print(f'{n:<5}{p[0]:15}{(p[1] + p[2]) / 2:>8}')

print('-' * 35)

while True:
    nota_aluno = int(input('Mostrar a nota de qual aluno? [999 interrompe]: '))
    if nota_aluno == 999:
        break
    print('-' * 35)
    print(f'As notas de \033[1m{lista[nota_aluno][0]}\033[m são {lista[nota_aluno][1::]}')
    print('-' * 35)

```

<https://youtu.be/7xrCJnniqMw>