

LISTA 3 – Programas com listas em Python – Parte 2

- 1) Crie uma lista com números inteiros de 1 a 10. Em seguida, use um loop para encontrar a soma de todos os números pares dessa lista.
- 2) Dada uma lista `nomes = ["Ana", "Beatriz", "Carlos", "Daniel", "Eduarda", "Fabio"]`, escreva um código para contar quantos nomes têm mais de cinco letras.
- 3) Escreva um programa que receba uma lista de números e retorne outra lista com os números elevados ao quadrado. Por exemplo, dado `[2, 3, 4]`, o código deve retornar `[4, 9, 16]`.
- 4) Crie uma lista `idades` com as idades de 10 pessoas. Escreva um código que divida essa lista em duas listas: uma com idades menores que 18 anos e outra com idades maiores ou iguais a 18 anos.
- 5) Dada uma lista de números inteiros `numeros = [3, 6, 9, 12, 15, 18]`, escreva um código para encontrar e remover o maior número da lista sem usar a função `max`.
- 6) Escreva uma função `remover_duplicatas(lista)` que receba uma lista de números e retorne uma nova lista sem elementos duplicados. Por exemplo, dado `[1, 2, 3, 1, 4, 2]`, a função deve retornar `[1, 2, 3, 4]`.
- 7) Crie uma lista `temperaturas = [23.5, 18.6, 30.2, 15.9, 25.1, 29.8]` e escreva um código para encontrar a média das temperaturas. Imprima a média ao final.
- 8) Escreva um programa que receba duas listas de números de mesmo tamanho e retorne uma nova lista onde cada elemento é a soma dos elementos nas mesmas posições nas listas originais. Exemplo: se `lista1 = [1, 2, 3]` e `lista2 = [4, 5, 6]`, o código deve retornar `[5, 7, 9]`.
- 9) Dada uma lista de palavras, escreva um programa que retorne uma nova lista contendo apenas as palavras que começam com uma vogal. Exemplo: se `palavras = ["gato", "elefante", "urso", "abelha", "cobra"]`, o programa deve retornar `["elefante", "urso", "abelha"]`.
- 10) Crie uma lista `notas = [7.5, 8.0, 9.0, 4.5, 6.0, 10.0, 5.0]` e escreva um programa que identifique a nota mais alta e a nota mais baixa da lista sem usar as funções `max` e `min`. Exiba ambas ao final.
- 11) Escreva uma função `intercalar_listas(lista1, lista2)` que receba duas listas de qualquer tamanho e retorne uma nova lista contendo os elementos intercalados de ambas. Se uma lista for maior, adicione os elementos restantes ao final da nova lista.
Exemplo: `lista1 = [1, 3, 5]`, `lista2 = [2, 4, 6, 8, 10]` -> `[1, 2, 3, 4, 5, 6, 8, 10]`
- 12) Crie uma função que receba uma lista de números e retorne uma nova lista contendo apenas os números que são a soma de dois outros números distintos na mesma lista.
Exemplo: `lista = [3, 5, 7, 10, 15]` -> `[10, 15]` (pois $10 = 3 + 7$ e $15 = 5 + 10$)

13) Escreva uma função `rotacionar_lista(lista, n)` que rotacione os elementos de uma lista `n` vezes para a direita.

Exemplo: `rotacionar_lista([1, 2, 3, 4, 5], 2)` -> `[4, 5, 1, 2, 3]`

14) Escreva um programa que receba uma lista de listas (matriz) e calcule a soma de todos os elementos em cada linha, retornando uma nova lista com essas somas.

Exemplo: `matriz = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]` -> `[6, 15, 24]`

15) Crie uma função que receba uma lista de números e retorne uma lista de tuplas, onde cada tupla contenha um número e sua frequência na lista. Ordene a lista de tuplas por frequência em ordem decrescente.

Exemplo: `lista = [4, 4, 3, 2, 4, 3]` -> `[(4, 3), (3, 2), (2, 1)]`

16) Escreva uma função `agrupa_por_paridade(lista)` que divida uma lista de números em duas listas: uma contendo os números pares e outra os ímpares, e retorne uma lista contendo ambas as listas.

Exemplo: `lista = [1, 2, 3, 4, 5, 6]` -> `[[2, 4, 6], [1, 3, 5]]`

17) Escreva um programa que, dada uma lista de listas (matriz), retorne outra matriz que seja a transposta da original.

Exemplo: `matriz = [[1, 2, 3], [4, 5, 6]]` -> `[[1, 4], [2, 5], [3, 6]]`

18) Crie uma função `menor_maior_diferenca(lista)` que receba uma lista de números e retorne a diferença entre o maior e o menor número da lista sem usar as funções `max` e `min`.

19) Dada uma lista de números, escreva um programa que retorne todos os pares de elementos que somam um valor específico `k`.

Exemplo: Para `lista = [1, 2, 3, 4, 5]` e `k = 6`, o programa deve retornar `[(1, 5), (2, 4)]`.

20) Escreva uma função que receba uma lista de números e retorne uma nova lista onde cada elemento é o produto de todos os outros números na lista, exceto ele mesmo (sem usar divisão).

Exemplo: Para `[1, 2, 3, 4]`, a função deve retornar `[24, 12, 8, 6]`.