

MODALIDADE:	Curso de Especialização Tecnológica (CET) – 01/2025	Não aplicável
CURSO:	Técnico/a Especialista em Automação, Robótica e Manutenção Industrial	
UC:	Desenvolver programas em linguagem estruturada	CÓDIGO UC: UC00606
FORMADOR/A:	Bruno Silva	DATA:

OBJETIVOS

- Exercícios de listas e subprogramas com a linguagem Python

Exercícios Listas

- Construa um programa para **criar uma variável do tipo lista com o nome “numeros”**. De seguida, faça os seguintes exercícios:
 - Adicione** 4 novos elementos numéricos inteiros com a função **append()**;
 - Adicione** 1 novo elemento **entre o índice 1 e 2** com a função **insert()**;
 - Altere** o valor do **índice 2** para **número a sua escolha**;
 - Remova o índice 1** com a função **pop()**;
 - Mostre quantos elementos existem** na lista com a função **len()**;
 - Percorra a lista de valores com uma estrutura de repetição FOR e mostre os dados do índice e valor que percorre no momento;
 - Complete o exercício anterior e use a função **enumerate()** para criar um contador automático à medida que percorre a estrutura de repetição;
- Construa um programa para **criar uma variável do tipo lista com o nome “cidades”**. De seguida, faça os seguintes exercícios:
 - Adicione** 4 novos elementos textuais com a função **append()**;
 - Adicione** 1 novo elemento **entre o índice 2 e 3** com a função **insert()**;
 - Altere** o valor do **índice 1** para **um valor textual a sua escolha**;
 - Remova o índice 3** com a função **pop()**;
 - Mostre quantos elementos existem** na lista com a função **len()**;
 - Percorra a lista de valores com uma estrutura de repetição FOR e mostre os dados do índice e valor que percorre no momento;
 - Complete o exercício anterior e use a função **enumerate()** para criar um contador automático à medida que percorre a estrutura de repetição;

3. Neste exercício vão ser fornecidas duas listas de Python com valores inteiros. Cada lista representa os gastos de 5 meses da Ana e do João:

gastos_ana = [310, 250, 210, 800, 550]

gastos_joao = [275, 410, 525, 321, 250]

O objetivo é criar duas variáveis para guardar a soma dos gastos de cada um ao longo dos 5 meses. Utilize a função **sum()** para somar os valores de cada lista e a seguir, deve utilizar uma estrutura de decisão dupla encadeada para verificar qual dos dois teve mais gastos ou se tiveram os mesmos gastos.

4. Neste exercício vamos ter um conjunto de temperaturas para analisar.

temperaturas = [18.5, 20.0, 19.3, 21.7, 22.1, 17.8, 16.9, 23.4, 24.0, 19.8]

O objetivo será elaborar um programa para:

- Utilizar a função **max()** e descobrir o **valor numérico mais alto**;
- Utilizar a função **min()** e descobrir o **valor numérico mais baixo**;
- Utilizar a função **sum()** e **somar todos os valores**;
- Utilizar a função **len()** e **contar quantos elementos existem numa lista**;
- Utilizar a função **mean()** e mostrar a **média** (mas precisa de importar o módulo **statistics**)

5. Elabora um programa para encontrar o segundo maior valor da lista mais abaixo:

numeros = [32, 10, 58, 30, 55, 12, 28, 51]

Dica: primeiro faça a ordenação da lista com a função **sort()**. Como os valores foram ordenados, o segundo maior valor corresponderá ao penúltimo elemento da lista, e podemos utilizar o valor da indexação para obter o valor de uma posição, como por exemplo, **numeros[-2]**.

Exercícios Subprogramas

6. Construa um programa para **criar uma função com o nome “Mostra_Nome” e receba 2 parâmetros de entrada: nome e idade. No subprograma**, deve mostrar os valores que recebeu no parâmetro de entrada.
7. Construa um subprograma para **criar uma função com o nome “Soma_Numeros” e receba 2 parâmetros de entrada. No subprograma**, deve somar os valores que recebeu nos parâmetros de entrada e mostrar o resultado da soma.

8. Construa um subprograma para **criar uma função com o nome “Calcula_Area_Triangulo” e receba dois parâmetros de entrada: base e altura. No subprograma**, deve calcular a área dos valores que recebeu nos parâmetros de entrada e mostrar o resultado.
9. Construa um subprograma para **criar uma função com o nome “Calcula_Media_Notas” e receba 3 parâmetros de entrada: nota1, nota2, nota3. No subprograma**, deve calcular a média das notas que recebeu nos parâmetros de entrada e mostrar o resultado.
10. Construa um subprograma para **criar uma função com o nome “Calcula_Saldo” e receba 2 parâmetros de entrada: saldo e levantar. No subprograma**, deve calcular a diferença entre os dois valores, mostrar o resultado.
- Se** o cálculo do valor for negativo deve exibir a mensagem “Não pode retirar valores abaixo de 0”, **senão**, deve mostrar o saldo final da operação.