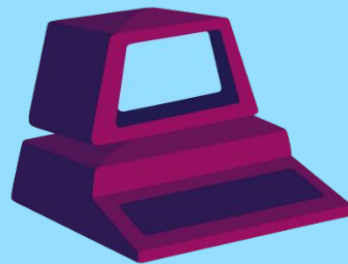


Aula 5 - Turma Avançado

- Listas

PET



COMPUTAÇÃO

2026

1.

Introdução às listas

Problema

- Imaginem que temos que criar uma variável para armazenar a idade de uma pessoa.
- Agora imaginem que precisamos armazenar as idades de 10 pessoas.
- E de 100, 1.000 ou 10.000 pessoas?

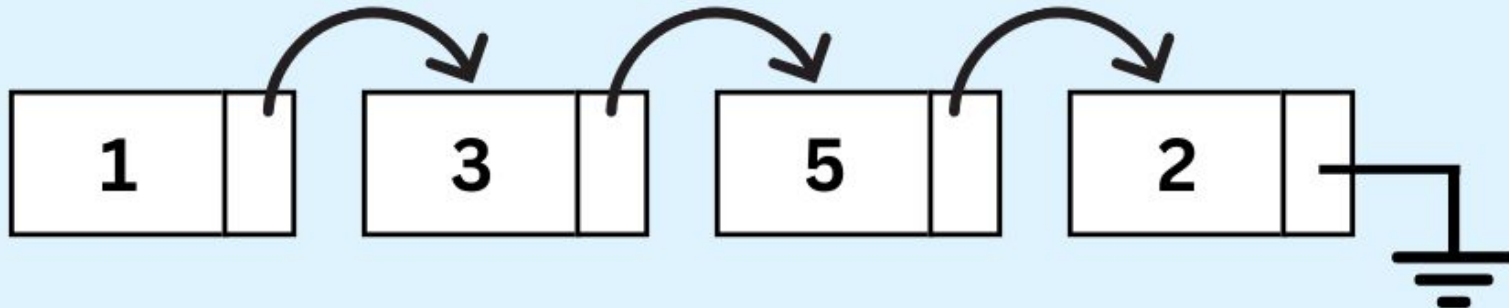
O que é uma lista?

- De forma simples, podemos pensar numa lista como uma coleção de itens.



O que é uma lista?

- Na programação, uma lista é uma estrutura de dados que permite armazenar essa coleção de elementos.



Declarando uma lista

- Em Python, declaramos uma lista usando um nome seguido de `=`. Para uma lista vazia, usamos `[]`. Para uma lista com valores, colocamos os elementos entre colchetes separados por vírgulas.
- **Obs.:** o exemplo abaixo usa um nome apenas ilustrativo, não recomendado na prática.



```
1 lista = []
```



```
1 lista = [1, 2, 3]
```

Acessando elementos de uma lista

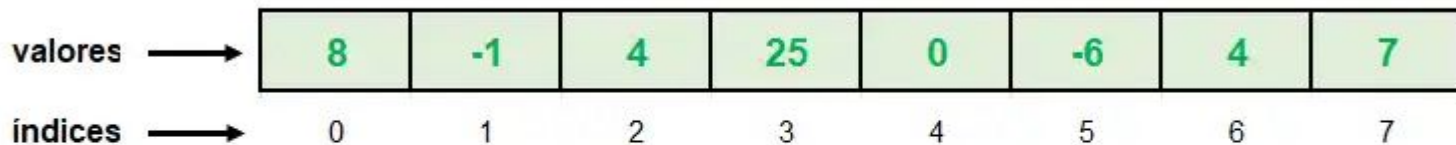
- Para acessar os elementos, utilizamos o identificador da lista seguido de colchetes e o índice do elemento que queremos acessar.



```
1 lista[0] = "Primeiro item"  
2 print(lista[0]) # Primeiro item
```

Índices de uma lista

- Cada elemento da lista possui um índice, através do qual podemos acessar ele.
- Apesar de parecer estranho, o índice do primeiro elemento da lista é 0, e não 1.



The diagram illustrates an array structure. It consists of a horizontal row of eight light green rectangular boxes, each containing a number. Below each box is its corresponding index, ranging from 0 to 7. To the left of the boxes, the word 'valores' (values) is followed by an arrow pointing to the first box (index 0). Below the boxes, the word 'índices' (indices) is followed by an arrow pointing to the first index (0).

valores	→	8	-1	4	25	0	-6	4	7
índices	→	0	1	2	3	4	5	6	7

Métodos e funções de listas

- Um método permite acessar ou modificar elementos de uma lista, ou a própria lista como um todo.
- Por exemplo, existem métodos para adicionar, remover, encontrar ou organizar elementos em uma lista.
- Já as funções podem ser usadas com listas, mas não pertencem diretamente a elas. Elas recebem a lista como argumento e retornam algum resultado, como o tamanho da lista.

Métodos e funções de listas



```
1  lista = [3, 1, 2, 1]
2  lista.append(5) # 3, 1, 2, 1, 5
3  lista.pop(4) # 3, 1, 2, 1
4  lista.sort() # 1, 1, 2, 3
5  lista.count(1) # 2
6  len(lista) # 4
7  min(lista) # 1
8  max(lista) # 3
9  sum(lista) # 7
```

Métodos e funções de listas

- Para ver saber mais sobre os métodos e funções das listas no Python, acesse

<https://docs.python.org/pt-br/3.13/tutorial/datastructures.html>

Problema 2

- E para inicializar uma lista com 10.000 posições?
- Teríamos que escrever 10.000 linhas de código???

2.

Listas e estruturas de repetição

Laços em lista

- Como já vimos anteriormente, as estruturas de repetição, laços ou loops (como **for** e **while**) nos permitem repetir determinadas instruções ou linhas de código.
- Além disso, usando loops podemos percorrer listas de maneira rápida e automática.

Percorrendo listas com laços



```
1  lista = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
2  for elemento in lista:
3      print(elemento, end=" ")
4
5  # 1 2 3 4 5 6 7 8 9 10
```

Função range()

- A função **range()** nos permite gerar uma sequência de números inteiros que é muito útil para trabalhar com listas e laços de repetição **for**.
-
- Estrutura: range(início, fim, passo)

Função range()

- A função **range()** nos permite gerar uma sequência de números inteiros que é muito útil para trabalhar com listas e laços de repetição **for**.
- Estrutura: **range**(início, fim, passo)



```
1  for i in range(1, 10, 1):  
2      print(i, end=" ")  
3  
4  # 1 2 3 4 5 6 7 8 9
```

Indentação

- *Lembrando:* o uso de indentação é **obrigatório** no Python

Código bem indentado



```
1  for i in range(1, 10, 1):  
2      print(i, end=" ")
```

Código mal indentado



```
1  for i in range(1, 10, 1):  
2  print(i, end=" ")
```

3.

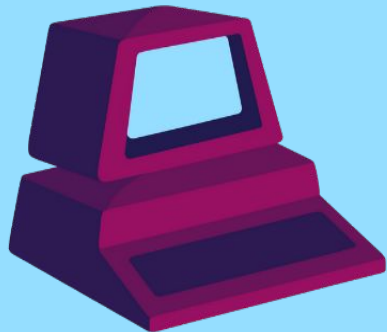
Exercícios

Exercícios

Usando **listas**

1. Crie uma lista com os números de 1 a 10 e imprima cada elemento dela
2. Dada uma lista de números, calcule e imprima a soma de todos os elementos.
3. Dada uma lista de números, encontre e imprima o maior e o menor valor
4. Dada uma lista de números, conte quantos são pares e quantos são ímpares.

Obrigado!



PET
COMPUTAÇÃO

pet.inf.ufpr.br
pet@inf.ufpr.br
@petcompufpr