Listas (continuação)

Percorrendo elementos de uma lista

Laço `for`:

- Ideal para percorrer elementos conhecidos.
- Exemplo:

```
for item in lista:

    print(item)
```

Percorrendo elementos de uma lista

Laço `while`:

- Usado quando não sabemos previamente o número de iterações.
- Exemplo:

```
i = 0

while i < len(lista):

    print(lista[i])

    i += 1
```

List comprehension

Uma maneira compacta e eficiente de criar listas.

List comprehension - sintaxe

```
nova_lista = [expressão for item in lista if condição]
```

List comprehension - exemplos

```
pares = [x for x in range(10) if x % 2 == 0]


quadrados = [x**2 for x in range(1, 11)]


letras = ["a", "b", "c", "d"]

maiúsculas = [letra.upper() for letra in letras]
```

Funções úteis

`len()`: Retorna o tamanho da lista.

`sum()`: Soma todos os elementos de uma lista numérica.

`min()` **e** `max()`: Encontram o menor e o maior elemento, respectivamente.

`sorted()`: Retorna uma nova lista ordenada (sem alterar a original).

Funções úteis

`enumerate()`: Permite iterar com índices e elementos ao mesmo tempo.

`zip()`: Combina duas listas em pares.

`reversed()`: Retorna um iterador para os elementos na ordem inversa.

Funções úteis

```python
nomes = ["João", "Maria", "Ana"]
idades = [25, 30, 22]


for i, c in enumerate(nomes):
    print(i, c)


for nome, idade in zip(nomes, idades):
    print(f"{nome} tem {idade} anos")
```

Funções úteis

`map(func, lista)`: Aplica uma função a todos os elementos de uma lista.

`filter(func, lista)`: Filtra os elementos da lista que atendem a uma condição.

Funções úteis

```
def dobra(x):
    return x * 2

dobrados = list(map(dobra, numeros))
print(dobrados)


Codeium: Refactor | Explain | Generate Docstring | ×
def eh_par(x):
    return x % 2 == 0


pares = list(filter(eh_par, numeros))
print(pares)
```

Desafio 01

Crie uma agenda de contatos inteligente

- Aplicativo de contatos com funcionalidades como:
  - Adicionar contato
  - Busca por nome/telefone
  - Organizar por grupos

Desafio 2

Análise de Temperaturas

Crie um programa que:

- Peça ao usuário para inserir as temperaturas dos últimos 7 dias
- Calcule e mostre:
  - A maior e menor temperatura
  - A temperatura média
  - Quantos dias ficaram acima da média
- **Desafio extra:** Mostrar um gráfico das temperaturas (no terminal)