



# <CodeCamp/> <@FEUP/>

14 fevereiro 2026





# Prepara o orçamento da tua festa

# <Listas/>

## Listas

Nesta secção vais fazer alguns exercícios relacionados com listas em Python.

Começa por criar uma lista lista chamada **compras** com 5 itens essenciais para a festa.

## Listas

Ao longo dos exercícios desta secção não alteres a linha que usaste para criar esta lista.

Utiliza as ferramentas de manipulação de listas para obter os resultados pedidos.



## Exercício 1

Dá print ao primeiro e ao último item da lista.



Lembrete: O acesso às listas é feito através dos parêntesis retos [ ].

## Exercício 2

Pede ao utilizador o nome de um novo item. De seguida, adiciona-o à tua lista de compras e dá print da lista atualizada.



Dica: Lembras-te da função que é usada para ler o input do utilizador?



## Exercício 3

Oops! Afinal o terceiro item da lista está esgotado...

Troca-o por outro e mostra a lista nova.



Dica: Lembras-te da função que é usada para ler o input do utilizador?

## Exercício 4

Conta quantos itens tens no total no carrinho.  
Dá print da frase: “Tens x artigos no carrinho.”



Se tiveres resolvido todos os exercícios corretamente até agora, deves ter 6 itens na tua lista.

# <Dicionários />

U.PORTO  
FEUP FACULDADE DE ENGENHARIA  
UNIVERSIDADE DO PORTO

FEUP  
**ESCOLAS**  
Programa de captação  
e desenvolvimento de talento.



## Dicionários

Os dicionários armazenam os dados em pares chave:valor. Esta estrutura de dados não permite itens duplicados, ou seja, não podes ter dois itens com a mesma chave.

## Dicionários

Um exemplo de um dicionário em Python:

```
1
2     cc = {
3         "name": "Alice",
4         "age": 30,
5         "city": "New York"
6     }
7
8     print(cc["name"]) # Output: Alice
9
```



## Dicionários

Funções básicas de dicionários em Python:

```
11
12  cc["age"] = 31 # Atualizar um valor
13  print(cc["age"]) # Output: 31
14
15  cc["country"] = "USA" # Adicionar um novo item
16  print(cc) # Output: {'name': 'Alice', 'age': 31, 'city': 'New York', 'country': 'USA'}
17
18  del cc["city"] # Remover um item
19  print(cc) # Output: {'name': 'Alice', 'age': 31, 'country': 'USA'}
20
```



## Exercício 5

Cria um dicionário chamado **preços** onde as chaves são os produtos e os valores são os preços respetivos.

Cria 5 artigos e, de seguida, dá print a um artigo chamado **leite** que custa **0.80**.



Lembra-te dos vários tipos de dados! **String** é o tipo ideal para texto, enquanto que **int** é o tipo ideal para valores numéricos.

## Exercício 6

Pede ao utilizador para escrever o nome e o preço de um produto e adiciona-o ao teu dicionário.  
De seguida dá print à seguinte string: “O artigo X custa Y euros.”



Dica: A função print pode receber vários parâmetros. Ex.: print("Isto é uma string ", variável, " outra string")



## Exercício 7

Parece que esgotou o stock do leite...  
Remove o artigo **leite** do teu dicionário!

 Dá print ao dicionário para confirmares que removeste o item com sucesso!



## Exercício 8

A inflação não nos larga. Está tudo a ficar mais caro...

Aumenta o preço de cada artigo em 25%



## Exercício 9

Está na hora de ir pagar. Calcula o preço total de todos os artigos no dicionário e dá print do resultado final!

# <Funções/>



## Exercício 10

Cria uma função **fatias\_necessarias** que recebe como argumentos o número de convidados. Sabendo que cada pessoa come 3 fatias, a função deve devolver o total de fatias necessárias.

 Não queres que nenhum dos teus amigos coma uma fatia a menos! Adiciona **import math** à primeira linha do teu ficheiro e tenta descobrir para que serve a função **math.ceil()** 

## Exercício 10

Cria uma função **dividir\_conta** que recebe dois valores: o preço total da festa e o número de amigos. A função deve devolver quanto é que cada amigo tem de pagar.

## <Exercícios extra />



## Exercício Extra 1

Cria um dicionário chamado **preços** que guarde pares **item:preço**. Faz um loop que percorre todos os artigos. Se um dos artigos custar mais de 20€, o programa deve imprimir um aviso.



Para iterares por um dicionário: `for item, preço in preços.keys():`



## Exercício Extra 2

Ganhaste um vale de desconto de **20€!**

Escreve um programa que percorra uma lista de preços e encontre quais são os dois valores que, somados, dão exatamente o valor do teu vale de desconto.



Valida o teu código com várias listas diferentes!



## Exercício Extra 3

Cria uma função chamada **limpar\_lista** que receba uma lista de convidados com nomes repetidos e devolva (return) uma nova lista apenas com os nomes únicos.



Cria uma lista auxiliar!