



# <CodeCamp/> <@FEUP/>

14 fevereiro 2026





# Sessão 2: Fui promovido. E agora?

<Sets/>





## Sets

Os sets não são ordenados (a ordem em que adicionas itens, não é a mesma em que são guardados). Os sets não podem conter dados que possam ser alterados (listas, tuplos, outros sets). Para além disso, não permitem valores duplicados!



## Pedaços de código:

```
set = {1,2,3,4}  
print(len(set)) # imprime o número de elementos no set (4)  
print(1 in set) # imprime True se 1 estiver no set  
print(5 not in set) # imprime True se 5 não estiver no set  
set.add(5) # adiciona 5 ao set  
print(set) # imprime {1,2,3,4,5}
```

```
set2 = {1,23,4}  
set.update(set2) # adiciona os itens do set2 no set  
print(set) # imprime {1,2,3,4,23}  
set.remove(23) # remove o 23 do set, se o item não existir dá erro  
print(set) # {1, 2, 3, 4, 5}  
set.discard(5) # discard vai remover o número se ele existir e, ao contrário do remove, não vai dar erro caso o elemento não exista
```



## Pedaços de código:

```
set3 = set2 & set # interseção dos conjuntos  
print(set3) # imprime {1,4}
```

```
set4 = set2 | set # une os conjuntos  
print(set4) # imprime a união dos conjuntos {1, 2, 3, 4, 23}
```

```
set5 = set2 ^ set # mantém os elementos que não estão em ambos os conjuntos  
print(set5) # imprime {2,3,23}
```

```
set6 = set2 - set # elementos que estão no set2 mas não no set  
print(set6) # imprime {23}
```



Os Recursos Humanos enviaram-te uma **lista de clientes da empresa**. No entanto, recebemos um alerta: alguns dos nossos clientes podem estar a negociar em simultâneo com outras empresas... Felizmente, os ASPA (Agentes Secretos da Programação Avançada) conseguiram acesso a uma **lista de clientes que está a negociar com uma empresa concorrente**. A tua missão: **descobrir** quais dos nossos **clientes** estão **envolvidos** nessas **negociações!** Damos-te uma dica: lembra-te dos **conjuntos de matemática**!

Exemplo:

```
nosso_cliente = {"Joana", "Pedro", "Henrique", "Carlos"}
```

```
clientes_doutras_empresas = {"Joana", "Elísio", "Henrique", "Santos"}
```

```
resultado = {"Joana", "Henrique"}
```

# <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** e também não permitem duplicados, ou seja, itens com a mesma chave. Para além disso têm ordem.



## Pedaços de código:

```
dic = {'Ana':2, 'Abel':3, 'Tiago':4, 'Pedro':5}  
  
print(len(dic)) # imprime o tamanho do dicionário, 4 (número de pares chave:valor)  
  
print(dic['Ana']) # imprime o valor correspondente à chave "Ana", neste caso, 2  
  
print(dic.keys()) # imprime as chaves do dicionário (Ana,Abel,Tiago,Pedro)  
  
print(dic.values()) # imprime os valores do dicionário (2,3,4,5)  
  
dic['António']=25 # adiciona o par 'António':25 ao dicionário  
  
print(dic) # {'Ana':2, 'Abel':3, 'Tiago':4, 'Pedro':5, 'António':25}  
  
dic['António']=5 # altera o valor da chave 'António' no dicionário  
  
print(dic) # {'Ana':2, 'Abel':3, 'Tiago':4, 'Pedro':5, 'António':5}  
  
del dic['António'] # apaga o par cuja a chave é o António  
  
print(dic) # {'Ana':2, 'Abel':3, 'Tiago':4, 'Pedro':5}
```



O CEO, por acidente, modificou todas as palavras-passe dos seus funcionários. Consegues ajudá-lo a recuperar as mesmas sabendo as modificações?

**Atenção!** Se o carácter não estiver presente no dicionário, o mesmo deve permanecer igual.

Pass Modificada: “ahdvvhvau”

Modificações: {'a':2, "v":9,"u":5}

Pass Original: "2hd9h9h25"

# <Loops/>

U.PORTO  
FEUP FACULDADE DE ENGENHARIA  
UNIVERSIDADE DO PORTO

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



## For Loops

- String:

```
string="Como iterar sobre uma string"  
for letra in string:  
    print(letra)
```

- Listas:

```
lista=[1,2,3]  
for item in lista:  
    print(item)
```

- Dicionários:

```
dic={"Ana":1,"Carlos":3,"Vera":6}  
for x in dic:  
    print(x) # imprime as chaves do dicionário  
for x in dic.values():  
    print(x) # imprime os valores das chave  
for x in dic.keys():  
    print(x) # imprime as chaves
```

## Nested For loops

Servem para iterar sobre estruturas complexas, como por exemplo de listas de listas, dicionários como as chaves são listas e muitas outras situações, entre outras estruturas.

```
dic={1:("Ana",'10/09/2023'),2:("Carlos",'10/10/2020',5)}  
for x in dic.values():  
    for j in x:  
        print(j)
```



A empresa quer fazer uma venda de itens eletrónicos em que os mesmos vão estar com 10% de desconto, cria uma lista com os itens e os seus preços de venda modificados.

### Exemplo:

Itens: [(pc,400,"Eletrónicos"),(aspirador,100,"Eletrónicos"),(laca, 20,"Produtos de beleza"),(cadernos,15, "Produtos escolares")]

Preços após o desconto: [(pc,360,"Eletrónicos"),(aspirador,90,"Eletrónicos"),(laca, 20,"Produtos de beleza"),(cadernos,15, "Produtos escolares")]



Existe uma sala de alta segurança na empresa, que apenas os funcionários com estas duas condições podem aceder {"Segurança", "Informática"}.

Através de um dicionário, descobre quem são os funcionários que têm acesso à sala!

### Exemplo:

Funcionários: {"Inês": {"Eletrotécnica", "Vendas"}, "Maria": {"Mecânica"}, "Mário": {"Informática", "Contabilidade"}, "Ana": {"Informática", "Segurança", "Recursos Humanos"}}

Resultado: ["Ana"]



Os recursos humanos querem mandar uma mensagem às pessoas que estão na empresa há mais de 5 anos.

Para esta tarefa recebes uma lista com funcionários (Name, Department, Years of Service) imprime uma mensagem para estas pessoas, por exemplo: “Obrigado (Nome) pelo teu serviço”.

Funcionários = [("Ana Silva", "Recursos Humanos", 8), ("Miguel Costa", "TI", 3), ("Sónia Rocha", "Vendas", 12), ("Inês Santos", "Engenharia", 5), ("Ricardo Pereira", "Marketing", 1), ("Carla Antunes", "Logística", 6), ("Afonso Dias", "TI", 4), ("Abel Ferreira", "Finanças", 10)]



Os recursos humanos querem averiguar quais funcionários trabalham para mais de um departamento.

Para esta tarefa recebes uma lista de funcionários (Name,[Departamentos]), retorna uma lista com os nomes dos funcionários.

Funcionários: [("Ana Silva", ["Recursos Humanos", "Vendas"]),(("Miguel Costa", ["TI"]),(("Sónia Rocha", ["Vendas"]),(("Afonso Dias", ["TI", "Logística", "Segurança"]),(("Inês Santos", ["Engenharia"]),(("Carla Antunes", ["Logística", "Finanças"]),(("Abel Ferreira", ["Finanças"])))]