

Listas, Tuplos, Dicionários e Conjuntos

Carlos J. Costa



Listas

- Uma lista armazena uma série de items numa ordem específica
- Pode-se aceder a cada item utilizando um indice ou através de ciclo
- As listas são mutáveis

```
# Construir uma lista
```

```
lista_compras = ['batatas', 'cenouras', 'bacalhau', 'couves']
```

- **Ou então**

```
lista_compras = list(('batatas', 'cenouras', 'bacalhau', 'couves'))
```

Listas

- Operações com listas

```
# Obter o primeiro elemento da lista  
lista_compras[0]
```

```
# Obter o último elemento da lista  
lista_compras[-1]
```

Listas

- Operações com listas

Iterar através de uma lista

```
for compra in lista_compras:
```

```
    print(compra)
```

Listas

- Adicionar items

```
# Adicionar items a lista
filmes = []
filmes.append('Vice')
filmes.append('Green Book')
filmes.append('Roma')
filmes.append('A Star Is Born')
print (filmes)
```

Listas

- **Remover Elementos**

```
filmes.pop()
```

```
filmes.remove('A Star Is Born')
```

```
del films[1]
```

```
print (filmes)
```

- **Apagar lista**

```
clear(filmes)
```

Listas

- Compreensão de lista

```
# List comprehensions
```

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

Lista

- **Fatiar a lista**

```
# obter os três primeiros elementos da lista
lista_compras = ['batatas', 'cenouras', 'bacalhau', 'couves']
tres_primeiros= lista_compras[:3]
print(tres_primeiros)
```


Lista

- Qual o resultado de?

```
compras = lista_compras
lista_compras.append("laranjas")
print(compras)
```

- Pois.. Não copia... Para copiar, faz-se:

```
# copiar uma lista
copia_lista_compras = lista_compras [:]
```

Tuplos

- Os tuplos são idênticos às listas, mas não podem ser modificados
- São designados *imutáveis*

```
novas_compras = ("bananas", "feijão", "arroz")  
print (novas_compras[1])  
novas_compras[0]="maçã"
```

- Qual o resultado?

Dicionários

- Os dicionários armazenam ligações entre peças de informação
- Cada item de um dicionário é um par chave-valor
- Os valores não são repetíveis

```
fruta = {1: 'laranja', 2: 'maça', 3: 'pera', 4: 'uva', 5: 'pessego' }
```

Dicionários

- adicionar um novo par chave-valor

```
fruta[10] = 'romã'
```

- Iterar através de par chave-valor

```
for chave , valor in fruta.items():  
    print('a fruta ' + str(chave) + ' é ' + valor)
```

Dicionários

- Iterar através da chave

```
for chave in fruta.keys():  
    print(str(chave) + ' é fruta')
```

- Iterar através dos valores

```
for valor in fruta.values():  
    print(valor + ' é fruta')
```

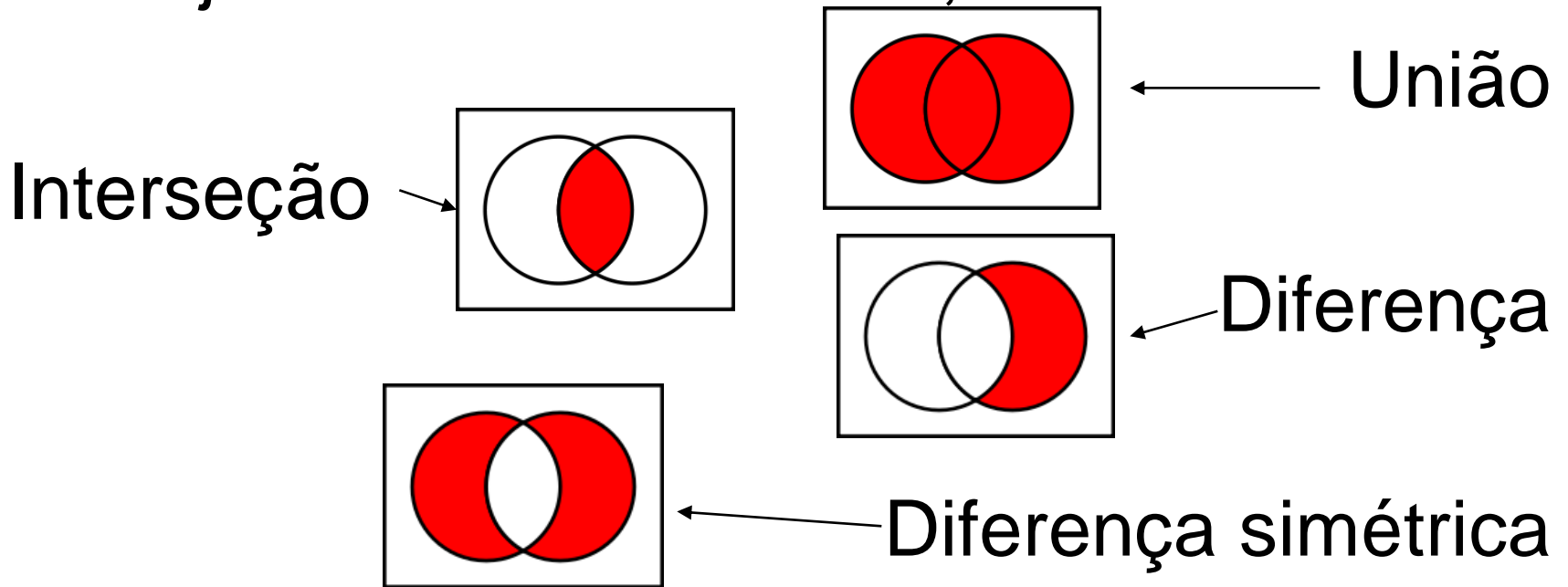
Conjuntos

- Sets (ou conjuntos)
- São estruturas disponíveis no Python, utilizadas para representar coleções não ordenadas de elementos únicos.

```
s = {1, 2, 3, 4}  
print (s)
```

Conjuntos

- Nos Conjuntos podem-se realizar um conjunto de operações típicas dos conjuntos da matemática, tais como:



Conjuntos

- `a = {1, 2, 3, 4}`
- `b = {3, 4, 5, 6}`
- `print (a.union(b))`
- `print (a.intersection(b))`
- `print (b.difference(a))`
- `print (a.symmetric_difference(b))`

Bibliografia

- <https://www.w3schools.com/python/>
- <https://www.python.org/>
- <https://docs.python.org/3/tutorial/>