Tuples

Os tuples em Python são uma estrutura de dados que permite armazenar uma coleção de dados ordenada. São semelhantes a arrays e listas mas são imutáveis. Isso significa que, uma vez criado, o conteúdo de um tuple não pode ser alterado.

Os tuples são uma coleção ORDENADA e IMUTÁVEL de elementos. Os elementos podem ser de qualquer tipo, incluindo outros tuples, e podem ser repetidos.

Sendo uma coleção imutável não é possível adicionar ou remover elementos de um tuple.

Declaração

In [ ]:
```
meu_tuple = (1,2,3)
print(meu_tuple)
#ou

meu_tuple_v2 = 1,2,3
print(meu_tuple_v2)
# ou

meu_tuple_v3 = tuple([1,2,3])
print(meu_tuple_v3)
#para declarar um tuplo com um único elemento

meu_tuple_v4 = (1,)
print(meu_tuple_v4)

#tuplo a partir de uma string
meu_tuple_v5 = tuple("olá mundo")
print(meu_tuple_v5)
```

Os elementos pode ser acedidos através de um índice, da mesma forma que os arrays ou as listas.

In [ ]:
```
frutas = ("maçã", "banana", "cereja")
print(frutas[0])
```

Os tuples são imutáveis, qualquer tentativa em alterar um elemento resulta em erro

In [ ]:
```
carros = ("ford","bmw","audi")
carros[0] = "VW"
```

Listar todos

In [ ]:
```
carros = ("ford","bmw","audi")
```

```
for carro in carros:
    print(carro)
```

In [ ]:
```
carros = ("ford","bmw","audi")

for i in range(len(carros)):
    print(carros[i])
```

Empacotamento e desempacotamento

In [ ]:
```
# empacotamento de um tuple
tuplo = 1, 2, 3

# desempacotamento
a, b, c = tuple

print(a,b,c)
```

Existe?

In [ ]:
```
carros = ("ford","bmw","audi")

if "ford" in carros:
    print("Existe um ford")
```

Métodos de tuples

Número de elementos de um tuple

In [ ]:
```
t = (1,2,3,4,5,6,2)

#número de elementos
print(len(t))

#número de ocorrencias de um elemento
print(t.count(2))

#índice da primeira ocorrência de um valor
print(t.index(3))
```

Utilidade dos tuples

Os tuples são úteis para devolver vários valores de uma função.

In [ ]:
```
def DevolveVarios():
    a = 1
    b = 2
    c = 3
    tuplo = a, b, c
    return tuplo

print(DevolveVarios())
```

```
x, y, z = DevolveVarios()
print(x,y,z)
```

Operadores com tuples

Os tuples suportam o operador + (soma) para concatenar dois tuplos e operador e o operador * (produto) para repetir um tuple

In [ ]:
```
tupla1 = (1, 2, 3)
tupla2 = (4, 5, 6)
resultado = tupla1 + tupla2  # resultado será (1, 2, 3, 4, 5, 6)
print(resultado)

triplo = tupla1 * 3
print(triplo)
```

Outras funções

Os tuples podem ser manipulados através do slicing.

Com os tuples podemos utilizar funções como min() e max() para encontrar o menor e o maior valor.

In [ ]:
```
numeros = (9,8,7)
menor=min(numeros)
maior=max(numeros)
print(menor,maior)
```

In [ ]:
```
numeros = (10,11,12)
soma = sum(numeros)
print(soma)
```

Também funciona com strings

In [ ]:
```
nomes = ("maria","António","carla")

print(min(nomes))
print(max(nomes))
```

Para obter os valores ordenados existe a função sorted()

In [ ]:
```
numeros = (9,8,7)

numeros_ordenados=sorted(numeros)
print(numeros_ordenados)

nomes = ("maria","António","carla")
nomes_ordenados=sorted(nomes)
print(nomes_ordenados)
```

Não é possível utilizar a função sort

In [ ]:
```
numeros = (9 ,15 ,1)
```

```
numeros.sort() #esta linha dá erro!
```