

# Tuplas em Python

---

# Tuplas

Tupla é um tipo de estrutura de dados do Python que tem como principal característica ser imutável. Ou seja, ao criar uma tupla você não consegue alterar nenhum elemento que faça parte dela.

Normalmente, as tuplas são utilizadas para armazenar sequências de códigos que não serão modificadas. Isso porque a sua característica de imutabilidade dão mais segurança ao desenvolvedor e maior proteção ao código.

# Tuplas

```
# Criando uma tupla vazia
tupla_vazia = ()

# Criando uma tupla com elementos
tupla_com_elementos = (1, 2, "três", 4.5, True)
```

Indexação e Acesso a Elementos: Os elementos de uma tupla podem ser acessados por meio de seus índices, que começam em 0, assim como em listas e outros tipos sequenciais em Python.

# Tuplas

```
tupla = (10, 20, 30, 40)

# Acessando elementos da tupla
print(tupla[0]) # saída: 10
print(tupla[2]) # saída: 30
```

Como mencionado anteriormente, as tuplas são imutáveis, o que significa que você não pode modificar os elementos individuais depois de criar a tupla. Por exemplo, você não pode adicionar, remover ou alterar elementos.

# Tuplas

```
tupla = (1, 2, 3)
# A tentativa de modificar a tupla resulta em erro
tupla[0] = 10 # TypeError: 'tuple' object does not support item assignment
```

Para saber quantos elementos uma tupla possui, você pode usar a função `len()`:

```
tupla = (10, 20, 30)
print(len(tupla)) # saída: 3
```

# Tuplas

Você pode concatenar duas ou mais tuplas usando o operador de adição +.

```
tupla1 = (1, 2, 3)
tupla2 = (4, 5, 6)
tupla_concatenada = tupla1 + tupla2
print(tupla_concatenada) # Output: (1, 2, 3, 4, 5, 6)
```

É possível repetir os elementos de uma tupla utilizando o operador de multiplicação \*.

```
tupla = (1, 2)
tupla_repetida = tupla * 3
print(tupla_repetida) # Output: (1, 2, 1, 2, 1, 2)
```

# Tuplas

Métodos de Tuplas: Como as tuplas são imutáveis, elas têm apenas dois métodos principais: `count()` e `index()`. O método `count()` conta o número de ocorrências de um elemento na tupla, e o método `index()` retorna o índice da primeira ocorrência de um elemento específico.

```
tupla = (10, 20, 30, 20)
print(tupla.count(20))  # Output: 2
print(tupla.index(30))  # Output: 2
```

# Tuplas

As tuplas são úteis quando você precisa garantir que os dados não sejam alterados acidentalmente ou quando deseja criar chaves de dicionário a partir de elementos imutáveis, entre outros casos. Lembre-se de que, se precisar de uma estrutura de dados mutável, você deve usar uma lista em vez de uma tupla.



# Exercícios de Fixação

Escreva um programa que crie uma tupla com 5 números inteiros e exiba os elementos da tupla na tela.

Crie um código que receba uma tupla e retorne a soma de todos os elementos.

Escreva um programa que crie duas tuplas e concatene-as em uma terceira tupla.

Crie um código que receba uma tupla de strings e retorne a quantidade de palavras que começam com a letra "A".

# Exercícios de Fixação

Escreva um programa que crie uma tupla com 10 números e informe qual é o maior e o menor valor presente na tupla.

Crie um código que receba uma tupla de números e retorne outra tupla com os elementos em ordem inversa.

Escreva um programa que leia uma sequência de números inteiros separados por espaço e armazene-os em uma tupla. Em seguida, exiba os números pares dessa tupla.

# Exercícios de Fixação

Escreva um programa que leia uma lista de nomes e suas respectivas idades e armazene-os em uma tupla. Em seguida, exiba os nomes das pessoas maiores de idade (idade  $\geq 18$ ).

Escreva um programa que leia uma lista de compras e seus respectivos preços, armazene-os em tuplas e exiba o total gasto.

Escreva um programa que leia uma tupla de strings e exiba a concatenação de todas as palavras.