

Aula 13

Tuplas

As tuplas em Python são uma estrutura de dados semelhante às listas, mas com algumas diferenças fundamentais. Enquanto as listas são mutáveis, as tuplas são imutáveis, ou seja, uma vez que uma tupla é criada, não é possível modificá-la. As tuplas também são definidas usando parênteses, enquanto as listas usam colchetes.

Para criar uma tupla em Python, basta colocar os valores desejados entre parênteses, separados por vírgulas. Por exemplo, para criar uma tupla de três elementos com os valores 1, 2 e 3, pode-se fazer o seguinte:

```
tupla = (1, 2, 3)
```

Também é possível criar uma tupla vazia usando um par de parênteses vazios:

```
1 tupla = ()
```

Uma vez criada, uma tupla pode ser acessada usando indexação, da mesma forma que uma lista. O primeiro elemento da tupla tem o índice 0, o segundo tem o índice 1 e assim por diante. Por exemplo, para acessar o segundo elemento da tupla "tupla" criada acima, pode-se fazer o seguinte:

```
1 tupla = (1, 2, 3)
2
3 print(tupla[1])
```

Também é possível usar a indexação negativa para acessar os elementos da tupla a partir do final. O último elemento tem índice -1, o penúltimo tem índice -2 e assim por diante.

```
1 tupla = (1, 2, 3)
2
3 print(tupla[-1])
4 print(tupla[-2])
5 print(tupla[-3])
```

Uma das principais vantagens das tuplas é que elas podem ser usadas como chaves em dicionários, enquanto as listas não podem. Isso ocorre porque as chaves em um dicionário devem ser imutáveis, e as tuplas são imutáveis.

Mais exemplos:

```
1 tupla = 123456, 654321, 'bom dia!'
2 print(tupla[0])
3 print(tupla)
4
5 # tuplas também podem ser aninhadas
6 outra_tupla = tupla, (1, 2, 3, 4, 5)
7 print(outra_tupla)
8
9 # tuplas são imutáveis
10 # TypeError: 'tuple' object does not support item
    assignment
11 tupla[0] = 8888
12
13 # mas elas podem conter objetos mutáveis
14 tupla = ([1, 2, 3], [3, 2, 1])
15 print(tupla)
```

Como você pode ver no trecho acima, na saída do console as tuplas são sempre envolvidas por parênteses, assim tuplas aninhadas podem ser lidas corretamente. Na criação, tuplas podem ser envolvidas ou não por parênteses, desde que o contexto não exija os parênteses (como no caso da tupla dentro de uma expressão maior). Não é possível atribuir itens individuais de uma tupla, contudo é possível criar tuplas que contenham objetos mutáveis, como listas.

Apesar de tuplas serem similares a listas, elas são frequentemente utilizadas em situações diferentes e com propósitos distintos. Tuplas são imutáveis, e usualmente contém uma sequência heterogênea de elementos que são acessados via desempacotamento (veremos mais a seguir) ou índices. Listas são mutáveis, e seus elementos geralmente são homogêneos e são acessados iterando sobre a lista.

Um problema especial é a criação de tuplas contendo 0 ou 1 itens: a sintaxe usa certos truques para acomodar estes casos. Tuplas vazias são construídas por um par de parênteses vazios; uma tupla unitária é construída por um único valor e uma vírgula entre parênteses (não basta colocar um único valor entre parênteses). Feio, mas funciona. Por exemplo:

```
1 vazio = ()
2 unico = 'buenas', # repare na vírgula ao final
3 print(len(vazio))
4 print(len(unico))
5 print(unico)
```

F-strings

f-strings são uma forma de interpolação de strings em Python 3.6 ou posterior. Essa sintaxe permite que os desenvolvedores possam incluir valores de variáveis em uma string de forma mais simples e legível.

A sintaxe das f-strings é simples: basta prefixar uma string com a letra "f" e, em seguida, incluir as variáveis desejadas entre chaves "{}" dentro da string. Por exemplo:

```
1 nome = "Gandalf"
2 idade = 10000
3 mensagem = f"Olá, meu nome é {nome} e tenho {idade} anos."
4
5 print(mensagem)
```

Nesse exemplo, a string "mensagem" contém uma interpolação de variáveis, com o nome e idade do indivíduo sendo incluídos na string usando a sintaxe das f-strings.

As f-strings são úteis porque permitem a inclusão de expressões dentro das chaves "{}", permitindo operações matemáticas, chamadas de função e outras expressões complexas dentro de uma string. Além disso, elas tornam o código mais legível e menos suscetível a erros de formatação de strings, que eram comuns em métodos de interpolação de string anteriores, como o método `format()`.

Exercícios para Praticar

1. Crie uma tupla com três números inteiros e imprima a tupla.
2. Crie uma tupla vazia e imprima-a.
3. Crie uma tupla com um único elemento e imprima-a.
4. Crie uma tupla com cinco strings e imprima o terceiro elemento da tupla.
5. Crie uma tupla com três números inteiros e imprima o maior valor da tupla.
6. Crie uma tupla com quatro elementos e inverta a ordem dos elementos.
7. Crie uma tupla com seis elementos e remova o terceiro elemento.
8. Crie duas tuplas, cada uma com três elementos, e combine-as em uma única tupla.
9. Crie uma tupla com cinco números inteiros e calcule a soma dos elementos da tupla.
10. Crie uma tupla com quatro strings e verifique se uma determinada string está presente na tupla.
11. Crie uma tupla com três elementos e converta-a em uma lista.
12. Crie uma tupla com cinco números inteiros e encontre o índice do primeiro elemento igual a um determinado valor.
13. Crie uma tupla com quatro elementos e ordene os elementos em ordem crescente.
14. Crie uma tupla com seis elementos e calcule a média dos valores da tupla.
15. Crie uma tupla com três elementos e calcule o produto dos elementos da tupla.
16. Crie uma tupla com sete elementos e verifique se um determinado elemento está presente na tupla.
17. Crie uma tupla com quatro strings e verifique se todas as strings na tupla têm o mesmo comprimento.
18. Crie uma tupla com seis elementos e imprima todos os elementos da tupla, exceto o primeiro e o último.
19. Crie uma tupla com cinco números inteiros e encontre o menor valor da tupla.
20. Crie uma tupla com três elementos e calcule a soma dos dois primeiros elementos da tupla.
21. Crie uma tupla com cinco números inteiros e calcule o produto dos elementos pares da tupla.
22. Crie uma tupla com três tuplas internas, cada uma contendo um nome e uma idade, e imprima o nome da pessoa mais velha.
23. Crie uma tupla com cinco elementos e verifique se todos os elementos da tupla são diferentes.
24. Crie uma tupla com sete elementos e ordene os elementos em ordem decrescente.
25. Crie uma tupla com três strings e crie uma nova tupla com as mesmas strings, mas em ordem alfabética.
26. Crie uma tupla com seis números e crie uma nova tupla com os mesmos números, mas em ordem reversa.
27. Crie uma tupla com três números e calcule a soma dos três números usando apenas uma linha de código.
28. Crie uma tupla com cinco elementos e imprima o segundo e o quarto elemento da tupla.
29. Crie uma tupla com quatro strings e verifique se uma determinada string é igual a qualquer uma das strings na tupla.
30. Crie uma tupla com seis números inteiros e verifique se todos os elementos da tupla são ímpares.
31. Crie uma tupla com quatro elementos e crie uma nova tupla contendo apenas os elementos pares da tupla original.
32. Crie uma tupla com cinco elementos e calcule a média dos valores dos elementos da tupla, arredondando o resultado para duas casas decimais.
33. Crie uma tupla com sete elementos e verifique se pelo menos um elemento da tupla é negativo.
34. Crie uma tupla com três strings e crie uma nova tupla com as mesmas strings, mas em ordem inversa.
35. Crie uma tupla com seis elementos e calcule a soma dos valores dos elementos da tupla que são maiores do que 10.
36. Crie uma tupla com cinco elementos e imprima a tupla em ordem inversa, sem alterar a ordem dos elementos da tupla original.
37. Crie uma tupla com quatro números inteiros e verifique se pelo menos dois elementos da tupla são iguais.
38. Crie uma tupla com sete elementos e verifique se pelo menos um elemento da tupla é igual a uma determinada string.
39. Crie uma tupla com cinco elementos e calcule a média dos valores dos elementos da tupla.
40. Crie uma tupla com seis elementos e verifique se todos os elementos da tupla são menores do que um determinado valor.
41. Crie uma f-string que interpola o valor de uma variável chamada "nome" em uma string de saudação.
42. Crie uma f-string que interpola o valor de duas variáveis chamadas "nome" e "idade" em uma string.
43. Crie uma f-string que interpola o resultado da soma de duas variáveis chamadas "x" e "y".

44. Crie uma f-string que interpola o resultado da multiplicação de duas variáveis chamadas "x" e "y".
45. Crie uma f-string que interpola uma chamada de função que recebe um parâmetro chamado "x".
46. Crie uma f-string que interpola o valor de uma lista chamada "lista".
47. Crie uma f-string que interpola o valor de uma variável chamada "preco" com duas casas decimais.
48. Crie uma f-string que interpola o valor de uma variável chamada "preco" com duas casas decimais e um símbolo de moeda no início.
49. Crie uma f-string que interpola o valor de uma variável chamada "texto" com as primeiras cinco letras.
50. Crie uma f-string que interpola o valor de uma variável chamada "texto" com as últimas cinco letras.
51. Crie uma f-string que interpola o valor de uma variável chamada "texto" com as primeiras e últimas cinco letras.
52. Crie uma f-string que interpola o valor de uma variável chamada "lista" e formate os itens da lista em colunas separadas por tabulações.
53. Crie uma f-string que interpola o valor de uma variável chamada "numero" com uma vírgula separando cada grupo de três dígitos (ex: 1,000,000).
54. Crie uma f-string que interpola o valor de uma variável chamada "texto" com todas as letras em maiúsculo.
55. Crie uma f-string que interpola o valor de uma variável chamada "texto" com todas as letras em minúsculo.
56. Crie uma f-string que interpola o valor de uma variável chamada "texto" com cada palavra começando com uma letra maiúscula.
57. Crie uma f-string que interpola o valor de uma variável chamada "texto" com o número de caracteres na string.