**Dicionários**

Um dicionário é uma coleção, assim como as listas e as tuplas. Porém, enquanto as tuplas eram indexadas por um índice, os dicionários são indexados por chaves. Todo elemento em um dicionário possui uma chave e um valor. Chaves tipicamente são strings, valores podem ser qualquer tipo de dado.

# O dicionário é definido pelos símbolos { e }

dicionario = {}

# O dicionário não possui um "append".

# Adicionamos valores diretamente:

dicionario['cat'] = 'gato'

dicionario['dog'] = 'cachorro'

dicionario['mouse'] = 'rato'

print(dicionario)

print(type(dicionario))

'''

Saída:

{'cat': 'gato', 'dog': 'cachorro', 'mouse': 'rato'}

<class 'dict'>

'''

# Dicionários, assim como as listas, são mutáveis:

dicionario['dog'] = 'cão'

print(dicionario)

# Saída: {'cat': 'gato', 'dog': 'cão', 'mouse': 'rato'}

# Podemos criar o dicionário diretamente também:

dicionario2 = {'Curso': 'Python Pro', 'Linguagem':'Python', 'Módulo':2}

print(dicionario2)

# Saída: {'Curso': 'Python Pro', 'Linguagem': 'Python', 'Módulo': 2}

# Podemos utilizar o operador "in" para verificar se uma chave existe:

if 'cat' in dicionario:

print('cat existe!') # Sim

if 'bird' in dicionario:

print('bird existe!') # Não

if 'gato' in dicionario:

print('gato existe!') # Não

'''

Também podemos utilizar as funções .keys() e .values() para obter listas

com apenas as chaves ou apenas os valores do dicionário.

'''

chaves = dicionario.keys()

print(chaves)

# Saída: dict\_keys(['cat', 'dog', 'mouse'])

valores = dicionario.values()

print(valores)

# Saída:dict\_values(['gato', 'cão', 'rato'])

# Já a função .items(), retorna uma lista de tuplas (chave, valor) de um dicionário

itens = dicionario.items()

print(itens)

# Saída:dict\_items([('cat', 'gato'), ('dog', 'cão'), ('mouse', 'rato')])