

Computação I - Python

Dicionários

Apresentado por: Thais Luca Marques de Almeida

Produção DCC-UFRJ

Metodologia de referência <https://doi.org/10.5753/wei.2016.9683>



Considere que você precisa fazer uma função que guarde o nome e o telefone de seus amigos. Sua função também deve permitir a consulta aos telefones das pessoas.

Como guardar estas informações (nome e telefone)?

Como recuperar o telefone de uma dada pessoa?

Dicionário

Dicionários são estruturas para armazenar dados. Não são sequências como strings, listas e tuplas.

São **mapeamentos** formados por pares de *chave* — *valor*.

Chave1 → Conteúdo1

Chave2 → Conteúdo2

Chave3 → Conteúdo3

...

Representam uma coleção não ordenada de **valores** onde cada valor é referenciado através de sua **chave**.

Notação: { chave1: conteúdo1, chave2: conteúdo2, ..., chaveN: conteúdoN }

Dicionário

Dicionários são **mapeamentos** formados por pares de *chave* — *valor*.

As **chaves** funcionam como os índices de uma lista

Chave1 → Conteúdo1

Chave2 → Conteúdo2

Chave3 → Conteúdo3

...

As **chaves de dicionários** são dados de tipo imutável, geralmente strings (podem ser tuplas ou tipos numéricos).

Os valores em um dicionário são dados quaisquer.

Dicionário

```
1 >>> telefones = {"Carlos": "2222-2223", "Andre": "2121-9092", "Jose": "9999-9291"}
2
3 >>> telefones["Andre"]
4 '2121-9092'
5
6 >>> telefones["Jorge"]
7 Traceback (most recent call last):
8   File "<pyshell#8>", line 1, in <module>
9     telefones["Jorge"]
10  KeyError: 'Jorge'
11
12 >>> "Jorge" in telefones
13 False
14
15 >>> len(telefones)
16 3
```

Dicionário

```
1 >>> telefones = {"Carlos": "2222-2223", "Andre": "2121-9092", "Jose": "9999-9291"}
2
3 >>> "Jorge" in telefones
4 False
5
6 >>> telefones["Jorge"] = "8586-9091"
7
8 >>> telefones["Jorge"]
9 '8586-9091'
10
11 >>> telefones
12 {'Andre': '2121-9092', 'Jorge': '8586-9091', 'Jose': '8799-0405',
13  'Carlos': '2222-2223'}
```

- **Para adicionar um novo par chave:valor**

Perceba que, diferentemente de listas, atribuir a um elemento de um dicionário não requer que uma posição exista previamente.

- O dicionário não fornece garantia de que as chaves estarão ordenadas, mas a ordem em que os elementos aparecem não é importante, pois os valores são acessados somente através de suas respectivas chaves, e não de suas posições.

Dicionário

```
1 >>> telefones = {"Carlos": "2222-2223", "Andre": "2121-9092", "Jose": "9999-9291"}
2
3 >>> telefones["Jose"]
4 '9999-9291'
5
6 >>> telefones["Jose"] = "8799-0405"
7
8 >>> telefones["Jose"]
9 '8799-0405'
```

Dicionários são mutáveis. Mesma sintaxe da inserção! Não é inserida outra chave com o mesmo nome e valor diferente, pois chaves são únicas.

Dicionário

```
1 >>> telefones
2 {'Andre': '2121-9092', 'Jorge': '8586-9091', 'Jose': '8799-0405',
3  'Carlos': '2222-2223'}
4
5 >>> "Jorge" in telefones
6 True
7
8 >>> del telefones["Jorge"]
9
10 >>> telefones
11 {'Andre': '2121-9092', 'Jose': '8799-0405',
12  'Carlos': '2222-2223'}
```

Podemos usar a função `del` para remover mapeamentos de um dicionário. O mapeamento será removido a partir da chave associada ao mesmo.

Autores

- **João C. P. da Silva** ▶ Lattes
- **Carla Delgado** ▶ Lattes
- **Ana Luisa Duboc** ▶ Lattes

Colaboradores

- **Anamaria Martins Moreira** ▶ Lattes
- **Fabio Mascarenhas** ▶ Lattes
- **Leonardo de Oliveira Carvalho** ▶ Lattes
- **Charles Figueiredo de Barros** ▶ Lattes
- **Fabício Firmino de Faria** ▶ Lattes

Computação I - Python

Dicionários

Apresentado por: Thais Luca Marques de Almeida

Produção DCC-UFRJ

Metodologia de referência <https://doi.org/10.5753/wei.2016.9683>