Dicionários

Alessandro Botelho Bovo

Dicionários

- Estruturas de dados baseadas em **pares chave–valor**.
- Cada **chave** é única e aponta para um **valor**.
- Sintaxe: `{chave: valor}`

```
pessoa = {"nome": "Ana", "idade": 25}
print(pessoa["nome"])
```

```
Ana
```

Vantagens dos dicionários

- Acesso rápido por chave.
- Organização de dados de forma mais legível.
- Flexibilidade: valores podem ser de qualquer tipo.
- Muito usados para representar registros e estruturas JSON.

Acesso em dicionários

- Por chave (gera erro se a chave não existir)

```
pessoa = {"nome": "Ana", "idade": 25}
print(pessoa["idade"])
```

```
25
```

```
print(pessoa["cidade"])
```

```
KeyError: 'cidade'
```

Acesso em dicionários

- Com `.get()` (evita erro se a chave não existir)

```
pessoa = {"nome": "Ana", "idade": 25}
print(pessoa.get("cidade"))
```

```
None
```

```
print(pessoa.get("cidade", "Não encontrado"))
```

```
Não encontrado
```

Iteração

- Pelas chaves

```
pessoa = {"nome": "Ana", "idade": 25}
for chave in pessoa:
    print(chave)
```

```
nome
idade
```

Iteração

- Pelos valores

```
pessoa = {"nome": "Ana", "idade": 25}
for valor in pessoa.values():
    print(valor)
```

```
Ana
25
```

Iteração

- Pela chave e valor

```
pessoa = {"nome": "Ana", "idade": 25}
for chave, valor in pessoa.items():
    print(chave, valor)
```

```
nome Ana
idade 25
```

Atualização com atribuição direta (`[]`)

- Atualiza **apenas uma chave por vez**.
- Se a chave **já existe**, o valor é **atualizado**.
- Se a chave **não existe**, ela é adicionada.

```
pessoa = {"nome": "Ana", "idade": 25}

pessoa["idade"] = 26                    # altera valor
pessoa["cidade"] = "Londrina"           # adiciona nova chave

print(pessoa)
```

```
{'nome': 'Ana', 'idade': 26, 'cidade': 'Londrina'}
```

Atualizar com .update()

- Pode atualizar **valores existentes** e adicionar **várias chaves** de uma vez.
- Se a chave **já existe**, o valor é **atualizado**.
- Se a chave **não existe**, ela é adicionada.

```
pessoa = {"nome": "Ana", "idade": 25}

pessoa.update({"idade": 27, "cidade": "Curitiba"})
```

```
{'nome': 'Ana', 'idade': 27, 'cidade': 'Curitiba'}
```

Resumindo a diferença

Forma	O que faz	Quando usar
`dicionario[chave] = valor`	- Atualiza o **valor** de uma chave existente. - Cria uma nova chave se ela não existir.	Quando for **apenas uma chave**.
`dicionario.update({...})`	- Atualiza os **valores de várias chaves existentes**. - Adiciona **novas chaves** de uma vez.	Quando quiser **atualizar/adicionar várias chaves ao mesmo tempo**.

- Para 1 chave → use `[]`.
- Para várias chaves → prefira `update()`.

Remoção

- O método `d.pop(chave)` **remove** a chave informada do dicionário e **retorna o valor** associado a ela.

- Se a chave **não existir**, ocorre um erro (`KeyError`), a menos que você forneça um **valor padrão**.

Método `d.pop(chave)`

- Removendo uma chave existente

```
pessoa = {"nome": "Ana", "idade": 25, "cidade": "Londrina"}

idade = pessoa.pop("idade")
print(idade)
print(pessoa)
```

```
25
{'nome': 'Ana', 'cidade': 'Londrina'}
```

Método `d.pop(chave)`

- Tentando remover chave inexistente (gera erro)

```
pessoa = {"nome": "Ana", "idade": 25}

# Isso gera KeyError, porque "curso" não existe
curso = pessoa.pop("curso")
```

```
in <module>
    curso = pessoa.pop("curso")
KeyError: 'curso'
```

Método `d.pop(chave)`

- Usando valor padrão para evitar erro

```
pessoa = {"nome": "Ana", "idade": 25}

curso = pessoa.pop("curso", "Não informado")
print(curso)
print(pessoa)
```

```
Não informado
{'nome': 'Ana', 'idade': 25}
```

Resumindo

- `pop(chave)` → remove e retorna o valor da chave.
- Se a chave não existir → erro (`KeyError`).
- Se passar um valor padrão → esse valor é retornado no lugar do erro.

Cuidados

- Chaves precisam ser imutáveis (`str`, `int`, `float`, `bool`, `tuple`).
- Valores podem ser mutáveis (`list`, `dict`).
- Acessar chave inexistente diretamente → gera erro.
- Use `.get()` para evitar.

Exemplos de dicionários

```
carrinho = {"arroz": 2, "feijão": 1, "macarrão": 3}

notas = {"Ana": 8.5, "João": 7.2, "Maria": 9.0}

funcionario = {"nome": "Carlos", "cargo": "Analista", "salario": 4500.00}

estoque = {"caneta": 100, "caderno": 50, "borracha": 80}

tradutor = {
    "cachorro": "dog",
    "gato": "cat",
    "casa": "house"
}
```

Dicionários aninhados

```
alunos = {
    "Ana": {"idade": 20, "curso": "SI", "nota": 8.5},
    "João": {"idade": 22, "curso": "Engenharia", "nota": 7.8},
    "Maria": {"idade": 19, "curso": "Química", "nota": 9.0}
}

estoque = {
    "frutas": {"maçã": 10, "banana": 20, "laranja": 15},
    "bebidas": {"água": 30, "refrigerante": 12},
    "higiene": {"sabão": 25, "shampoo": 8}
}

config = {
    "audio": {"volume": 70, "mudo": False},
    "video": {"brilho": 80, "contraste": 50},
    "rede": {"wifi": True, "modo_aviao": False}
}
```

Contato

Alessandro Botelho Bovo

alessandrobovo@utfpr.edu.br