



Dicionários: Criação e acesso aos dados

O que são Dicionários (`dict`) em Python?

Os **dicionários** são coleções de **pares chave:valor**!

Eles são úteis para armazenar dados organizados e permitir acesso rápido.

 **Diferença entre listas e dicionários:**

- **Listas** (`[]`): usam índices numéricos para acessar elementos.
- **Dicionários** (`{}`): usam **chaves únicas** para acessar valores.

1 Como Criar um Dicionário?

✓ Usando `{}` com pares **chave: valor**

```
peessoa = {  
    "nome": "Alice",  
    "idade": 25,  
    "cidade": "São Paulo"  
}  
print(peessoa)  
# Saída: {'nome': 'Alice', 'idade': 25, 'cidade': 'São Paulo'}
```

✓ Usando `dict()`

```
peessoa = dict(nome="Alice", idade=25, cidade="São Paulo")  
print(peessoa)  
# Saída: {'nome': 'Alice', 'idade': 25, 'cidade': 'São Paulo'}
```

2 Como Acessar Valores?

Podemos acessar os valores usando **as chaves**.

✓ Acessar um valor específico

```
print(pessoa["nome"])  
# Saída: Alice
```

✓ Usando `get()` (evita erro se a chave não existir)

```
print(pessoa.get("profissao", "Não informado"))  
# Saída: Não informado
```

3 Como Modificar um Dicionário?

✓ Alterar um valor

```
pessoa["idade"] = 26  
print(pessoa)  
# Saída: {'nome': 'Alice', 'idade': 26, 'cidade': 'São Paulo'}
```

✓ Adicionar uma nova chave

```
pessoa["profissao"] = "Engenheira"  
print(pessoa)  
# Saída: {'nome': 'Alice', 'idade': 26, 'cidade': 'São Paulo', 'profissao': 'Engenheira'}
```

✓ Remover uma chave

```
pessoa.pop("cidade")  
print(pessoa)  
# Saída: {'nome': 'Alice', 'idade': 26, 'profissao': 'Engenheira'}
```

✓ Remover o último item adicionado (`popitem()`)

```
pessoa.popitem()  
print(pessoa)  
# Saída: {'nome': 'Alice', 'idade': 26}
```

✓ Remover tudo (`clear()`)

```
peessoa.clear()
print(peessoa)
# Saída: {}
```

4 Percorrendo um Dicionário

✓ Percorrer chaves e valores

```
for chave, valor in pessoa.items():
    print(f"{chave}: {valor}")
```

📌 Saída:

```
nome: Alice
idade: 26
```

✓ Percorrer apenas as chaves

```
for chave in pessoa.keys():
    print(chave)
```

✓ Percorrer apenas os valores

```
for valor in pessoa.values():
    print(valor)
```

📌 Resumo Rápido

Operação	Código Exemplo	Saída
Criar dicionário	<code>d = {"chave": "valor"}</code>	<code>{'chave': 'valor'}</code>
Acessar valor	<code>d["chave"]</code>	<code>"valor"</code>
Acessar com <code>get</code>	<code>d.get("chave")</code>	<code>"valor"</code> ou <code>None</code>
Adicionar chave	<code>d["nova"] = 10</code>	<code>{'chave': 'valor', 'nova': 10}</code>
Alterar valor	<code>d["chave"] = 5</code>	<code>{'chave': 5}</code>

Remover chave	<code>d.pop("chave")</code>	<code>{'nova': 10}</code>
Remover último	<code>d.popitem()</code>	Remove último item
Limpar dicionário	<code>d.clear()</code>	<code>{}</code>
Percorrer chave	<code>for k in d.keys()</code>	Lista de chaves
Percorrer valor	<code>for v in d.values()</code>	Lista de valores
Percorrer tudo	<code>for k, v in d.items()</code>	Lista de pares