

Universidade Federal do Vale do São Francisco

Minicurso de Linguagem Python

**Edjair Aguiar Gomes Filho – Representante Discente pelo Centro Acadêmico de Engenharia
da Computação**

Aula 4 – Dicionários e Tuplas

Sumário

| | |
|--|----------|
| 1) DICIONÁRIO: DEFINIÇÃO | 2 |
| 2) DECLARANDO DICIONÁRIOS | 2 |
| 2.1) Utilizando {} | 2 |
| 2.2) Utilizando o método dict() | 2 |
| 3) OPERAÇÕES COM DICIONÁRIOS | 3 |
| 3.1) Retornando elementos pelo índice | 3 |
| 3.2) Tamanho de um dicionário..... | 3 |
| 3.3) Adicionar elementos | 3 |
| 3.4) Removendo elementos | 4 |
| 4) TUPLAS..... | 4 |
| 5) EXERCÍCIO..... | 5 |

1) DICIONÁRIO: DEFINIÇÃO

Em Python, os dicionários podem ser definidos como uma coleção de itens, sem ordem específica, onde cada elemento dentro do dicionário tem um identificador associado a ele. Esse identificador é chamado de *chave*. Esse tipo de estrutura é muito utilizado quando queremos, por exemplo, fazer bancos de dados pequenos e internos ao nosso programa, uma vez que cada dado vai ter uma identificação única.

É importante frisar que, como cada valor armazenado possui uma chave, as chaves devem ser únicas para que possamos acessar os elementos armazenados através delas.

Dessa forma, podemos enxergar o dicionário como uma coleção de pares na forma chave:valor. Assim como as listas, vistas na aula anterior, os dicionários têm tamanho variável e podem conter objetos de qualquer tipo, inclusive outros dicionários. Isso também funciona para as chaves: os identificadores também podem ser qualquer objeto.

2) DECLARANDO DICIONÁRIOS

Em ambos os casos abaixo, os elementos são separados das suas chaves com dois pontos (:) e separados entre si através da vírgula (,).

2.1) Utilizando {}

```
nome_do_dicionario = {1: 'João', 2: 'José'}  
nome_do_dicionario = {'nome': 'João', 'sobrenome': 'Silva'}
```

Figura 1 – Declaração de dicionário com {}

2.2) Utilizando o método dict()

Essa forma se diferencia da anterior apenas pelo uso do método dict(), que deve englobar a declaração do dicionário.

```
nome_do_dicionario_2 = dict({1: 'João', 2: 'José'})  
nome_do_dicionario_2 = dict({'nome': 'João', 'sobrenome': 'Silva'})
```

Figura 2 – Declaração de dicionário com o método dict()

3) OPERAÇÕES COM DICIONÁRIOS

3.1) Retornando elementos pelo índice

Para retornar um elemento pelo seu índice, podemos utilizar os colchetes ([]) ou o método get(). Em ambos os casos, passamos a chave do elemento que desejamos recuperar. Vamos declarar o seguinte dicionário:

```
dic_funcionarios = {'Operador' : 'Marcelo', 'Gerente' : 'Maria', 'RH' : 'João', 'Comercio' : 'José'}
```

Figura 3 – Dicionário de funcionários de uma empresa genérica

Podemos recuperar os nomes dos funcionários dessa empresa através dos seus cargos:

| | |
|--|-------|
| print(dic_funcionarios['RH']) | João |
| print(dic_funcionarios.get('Gerente')) | Maria |

3.2) Tamanho de um dicionário

Assim como para várias estruturas que já vimos anteriormente, o método len() novamente aparece por aqui para nos passar o tamanho de um dicionário.

| | |
|------------------------------|---|
| print(len(dic_funcionarios)) | 4 |
|------------------------------|---|

3.3) Adicionar elementos

Também existem duas formas. Na primeira, passamos o valor da nova chave e atribuímos um elemento a ela, da seguinte forma:

| | |
|---|--|
| dic_funcionarios['Financeiro'] = 'Renata' | |
| print(dic_funcionarios) | {'Operador' : 'Marcelo', 'Gerente' : 'Maria', 'RH' : 'João', 'Comercio' : 'José', 'Financeiro' : 'Renata'} |

A segunda forma é utilizando o método *update()*, passando como parâmetro a chave e o elemento a ser adicionado:

| | |
|---|--|
| <code>dic_funcionarios.update({'Publi' : 'Pedro'})</code> | |
| <code>print(dic_funcionarios)</code> | <code>{'Operador' : 'Marcelo', 'Gerente' : 'Maria', 'RH' : 'João', 'Comercio' : 'José', 'Financeiro' : 'Renata', 'Publi' : 'Pedro'}</code> |

3.4) Removendo elementos

Utilizamos o método *pop()* seguido da chave do elemento que deve ser removido.

| | |
|---|---|
| <code>dic_funcionarios.pop('RH')</code> | |
| <code>Print(dic_funcionarios)</code> | <code>{'Operador' : 'Marcelo', 'Gerente' : 'Maria', 'Comercio' : 'José', 'Financeiro' : 'Renata', 'Publi' : 'Pedro'}</code> |

4) TUPLAS

As tuplas, apesar de raramente mencionadas, são muito utilizadas em diversas aplicações. Tuplas são como listas, se diferenciando por ter imutável. Isso quer dizer que, após definida, a tupla não permite a adição ou remoção de elementos.

Porém é um equívoco pensar nas tuplas como listas constantes, uma vez que listas são destinadas a serem sequência homogêneas, enquanto tuplas são estruturas heterogêneas (Eby, 2005).

O que Phillip J. Eby quer dizer é: quando pensamos em uma lista, pensamos em organizar coisas do mesmo tipo. Pensamos numa lista de filmes, de livros, de séries, de disciplinas, de professores... Já quando pensamos em tuplas, devemos pensar em elementos de tipos distintos. Por exemplo, podemos fazer uma tupla que salve a data de aniversário de alguém:

```
alan_turing = ("23", "Junho", 1912)
```

Figura 4 – Tupla que contém dia, mês e ano

Daí estabelecemos outra diferença entre listas e tuplas: em uma lista, usamos colchetes para delimitar seus elementos, enquanto na tupla usamos os parênteses. Apesar disso, todos os outros métodos que estudamos sobre listas também se aplicam às tuplas, com exceção da adição e da remoção de elementos. Resumimos, então:

- Listas são delimitadas por colchetes, tuplas são delimitadas por parênteses;
- A ordem dos elementos de uma tupla não pode de alterada;
- Não é possível adicionar ou remover elementos de uma tupla já definida;
- A iteração, junção e a ideia de elementos como objetos são preservadas das listas para as tuplas;

5) EXERCÍCIO

Solicite ao usuário que insira os seguintes dados: nome, telefone, profissão e se o telefone é comercial ou doméstico. Crie um dicionário que vai funcionar como uma agenda telefônica, onde:

- A chave é o nome da pessoa;
- O elemento associado à chave é uma lista contendo telefone, profissão e tipo de telefone;

Em seguida imprima o item do dicionário no formato “chave: telefone-profissão-tipo”.