

# Python: Dicionários

**Galileu** Batista de Sousa  
Galileu.batista -at +ifrn -edu +br

# O que é um dicionário



- Um dicionário é um gerenciador de sinônimos.
  - "... pode fornecer, além das definições, informações sobre sinônimos, antônimos, ortografia, pronúncia, classe gramatical, etimologia etc."
- Em termos práticos:
  - É uma estrutura de dados que mapeia elementos
  - Dado um termo, retorna outro(s)
  - Correspondente a funções, mas com dados

# Fundamentos de Listas



- Listas são **coleções** de dados
- Podem conter vários elementos
  - Cada elemento pode ser de um tipo diferente

```
filhos    = ["Tiago", "Amanda", "Giovanna" ]  
sobrinha  = ["Bela", 10]  
irmaos    = [ ["Jr", 52], ["Riva", 43] ]
```

- Os elementos são acessados por índices: **0 .. len-1**

`sobrinha[0]` → `"Bela"`

# Fundamentos de Dicionários



- Dicionários também são **coleções** de dados
- Podem conter vários elementos
  - Cada elemento pode ser de um tipo diferente
  - Cada elemento tem uma **chave de acesso**

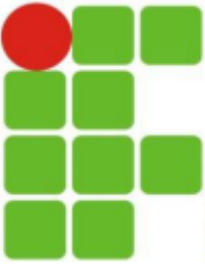
```
filhos    = { "filho": "Tiago", "f1": "Zi", "f2": "Gio" }  
sobrinha  = { "nome": "Bela", "idade": 10 }  
irmaos    = { 1: ["Jr", 52], 2: ["Riva", 43] }
```

sobrinha["nome"]



"Bela"

# Fundamentos de Dicionários



- São a estrutura de dados mais complexa do Python
- Enquanto listas demandam conhecimento da ordem dos elementos, dicionários não.
  - Não se pode garantir em que ordem estão guardados
- O acesso/definição é sempre feito pela chave.
  - Tecnicamente um dicionário é uma conjunto de pares:

Chave, valor

# Criação de Dicionários



- Três formas básicas:

- Dicionário vazio:

```
filhos = { }
```

- Dicionário vazio:

```
filhos = dict()
```

- Dicionário com valores:

```
sobrinhos = { "menina": "Bela", "menino": "Miguel" }
```

# Acesso a elementos



- Duas formas básicas

```
sobrinhos = { "menina": "Bela", "menino": "Miguel" }
```

- Indexação (a principal):

```
sobrinha = sobrinhos["menina"]  
print (sobrinha)
```

- Gera erro em caso de não existência da chave

- Método **get**

```
sobrinha = sobrinhos.get("M enina", "Não  
tem")
```

```
print (sobrinha)
```

- **Não** gera erro, retorna o valor *default*.

# Adição/alteração de elementos



- Adição pela atribuição a uma chave

```
bela = { "nome": "Isabela", "idade": 10 }  
bela [ "endereco" ] = "Brasilia"  
print (bela)
```

```
{ "nome": "Isabela", "idade": 10, "endereco" : "Brasilia" }
```

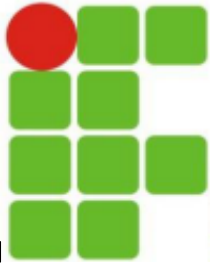
- Alteração pela atribuição a uma chave

```
bela [ "endereco" ] = "Quito"  
print (bela)
```

```
{ "nome": "Isabela", "idade": 10, "endereco" : "Quito" }
```



# Navegação em dicionários com for



- Dicionários são coleções de dados, certo?
  - Uso de **for** é adequado e suportado

```
filhos = { 1: "Tiago", 2: "Zi", 3: "Gio" }  
for pos in filhos:  
    print (filhos[pos])
```

Em cada repetição do **for** **pos** recebe uma chave do dicionário filhos

# Obtendo componentes de dicionários



- É possível obter uma tupla só com as chaves

```
filhos = { 1: "Tiago", 2: "Zi", 3: "Gio" }  
chaves = filhos.keys()  
print (chaves)  
for pos in chaves:  
    print (filhos[pos])
```

Não se vê diferença, pois o for,  
por padrão, usa `keys()`

```
(1, 2, 3 )  
Tiago  
Zi  
Gio
```

# Obtendo componentes de dicionários



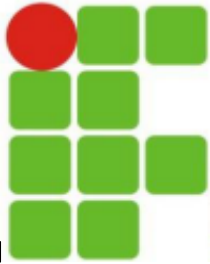
- É possível obter uma tupla só com as chaves

```
filhos = { 1: "Tiago", 2: "Zi", 3: "Gio" }  
for nome in filhos.values():  
    print (nome)
```

Aqui se vê diferença, pois o for,  
está usando **values()** do dicionário

Tiago  
Zi  
Gio

# Obtendo componentes de dicionários



- É possível obter uma lista com tuplas (chave, valor)

```
filhos = { 1: "Tiago", 2: "Zi", 3: "Gio" }  
pares = filhos.items()  
print (pares)  
for pos, nome in pares:  
    print (pos, nome)
```

Aqui o for está lendo um par  
**chave, valor** em cada iteração

```
[(1, "Tiago"), (2, "Zi"), (3, "Gio")]  
1 Tiago  
2 Zi  
3 Gio
```

# Dicionários podem ser complexos



- Acesse esse endereço:
  - <https://api.cartolafc.globo.com/clubes>
- Temos aqui um “mega” dicionário
  - Na verdade um mega string, com jeitão de dicionário
  - É o que chamamos de JSON
    - *Java Script Object Notation*
  - Usado em *web services*

# Conversão entre strings e dicionários



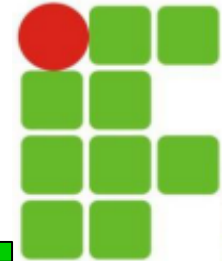
- Se um string tem o jeitão de dicionário, use:

```
import json
s = "{ '1': 'Ti', '2': 'Zi', '3': 'Gio' }"
dictS = json.loads(s)
```

- Para converter um dicionário em JSON

```
import json
d = { 1: 'Ti', 2: 'Zi', 3: 'Gio' }
sDict = json.dumps(d)
```

# Prática



' ' ' Procurar a nossa felicidade através da felicidade dos outros - aconselhava Olívia noutra carta sem data.

- Não estou pregando o ascetismo, a santidade, não estou elogiando o puro espírito de sacrifício e renúncia. Tudo isso seria inumano, significaria ainda uma fuga da vida. Mas o que procuro, o que desejo, é segurar a vida pelos ombros e estreitá-la contra o peito, beijá-la na face. Vida, entretanto, não é o ambiente em que te achas. As maneiras estudadas, as frases convencionais, o excesso de conforto, os perfumes caros e a preocupação do dinheiro são apenas uma péssima contrafação da vida. Buscar a poesia da vida será coisa que tenha nexos?

Ele agora vivia... Tinha tido apenas a ilusão de viver, mas na verdade andara morto por entre os homens.

O dia mais importante da minha vida foi aquele em que, recordando todos os meus erros, achei que já chegara a hora de procurar uma nova maneira de ser útil ao próximo, de dar novo rumo às minhas relações humanas. Que era que eu tinha feito senão satisfazer os meus desejos, o meu egoísmo? Podia ser considerada uma criatura boa apenas porque não matava, porque não roubava, porque não agredia? A bondade não deve ser uma virtude passiva. No dia em que eu achei Deus, encontrei a paz para mim e ao mesmo tempo percebi que de certa maneira não haveria paz para mim. Descobri que a paz interior só se conquista com o sacrifício da paz exterior. Era preciso fazer alguma coisa pelos outros. O Mundo está cheio de sofrimento, de gritos de socorro. Que tinha eu feito até então para diminuir esse sofrimento, para atender a esses apelos? Eu via em meu redor pessoas aflitas que, para se salvarem, esperavam apenas a mão que as apoiasse, nada mais que isso. E Deus dera-me duas mãos. Pensei tudo isso numa noite de insônia. Quando o dia nasceu, senti que tinha nascido de novo com ele. Era uma mulher nova.' ' '

Érico Veríssimo – Olhais os lírios do campo.

Faça um programa que conta quantas vezes cada palavra aparece no texto acima