

IMD0033 - Probabilidade

Aula 03 - Introdução a Python I

Ivanovitch Silva
Agosto, 2018



Agenda

- Fundamentos básicos de Python
- Arquivos e estrutura de repetição
- Estruturas condicionais
- Lista



crimes_rates.csv
dq_unisex_names.csv
la_weather.csv

Atualizar o repositório

```
git clone https://github.com/ivanovitchm/imd0033_2018_2.git
```

Ou

```
git pull
```

Fundamentos básicos de Python

```
import sys  
print(sys.version)
```

```
3.6.3 (default, Oct  3 2017, 21:45:48)  
[GCC 7.2.0]
```

Operators

Example expression

Addition

+

```
>>> 4 + 5  
9
```

Subtraction

-

```
>>> 4 - 5  
- 1
```

Multiplication

*

```
>>> 4 * 5  
20
```

Division

/

```
>>> 4 / 5  
0.8
```

Exponent

**

```
>>> 4 ** 5  
1024
```

Parentheses

()

```
>>> (4 ** 5)  
1024
```

Usando listas para almacenar múltiples valores

```
[ ] cities.append("Albuquerque")
    cities.append("Anaheim")
    print(cities)
    print(type(cities))
```

```
[> ['Albuquerque', 'Anaheim']
    <class 'list'>
```

Each time we call **list.append()**, the values in the **list cities** are updated.



Acessando elementos de uma lista

crime_rates					
index	0	1	2	3	4
values	749	371	828	503	1379

```
crime_rates = [749, 371, 828, 503, 1379]  
first_value = crime_rates[0]  
second_value = crime_rates[1]  
fifth_value = crime_rates[4]
```

Filtrar dados em uma lista

```
[ ] crime_rates = [749, 371, 828, 503, 1379]
    # The following slice selects values at index 2 and 3, but not 4.
    two_four = crime_rates[2:4]
    two_four
```

☞ [828, 503]

Here's a diagram of the same slice:

crime_rates					
index	0	1	2	3	4
values	749	371	828	503	1379

crime_rates[2:4]	
2	3
828	503

Montar o driver no google colab

```
[ ] #1. Install a Drive FUSE wrapper google-drive-ocamlfuse.  
!apt-get install -y -qq software-properties-common python-software-properties module-init-tools  
!add-apt-repository -y ppa:alessandro-strada/ppa 2>&1 > /dev/null  
!apt-get update -qq 2>&1 > /dev/null  
!apt-get -y install -qq google-drive-ocamlfuse fuse
```

SHOW HIDDEN OUTPUT

```
[ ] #2. Generate auth tokens for Colab  
from google.colab import auth  
auth.authenticate_user()
```

```
[ ] #3. Generate creds for the Drive FUSE library.  
from oauth2client.client import GoogleCredentials  
creds = GoogleCredentials.get_application_default()  
import getpass  
!google-drive-ocamlfuse -headless -id={creds.client_id} -secret={creds.client_secret} < /dev/null 2>&1 | grep URL  
vcode = getpass.getpass()  
!echo {vcode} | google-drive-ocamlfuse -headless -id={creds.client_id} -secret={creds.client_secret}
```

SHOW HIDDEN OUTPUT

```
[ ] #4. Create a directory and mount Google Drive using that directory.  
!mkdir -p drive  
!google-drive-ocamlfuse drive
```

```
[ ] import os  
os.chdir("/content/drive/Atividades/Ensino/Disciplinas/GRADUAÇÃO/Ciencias dos Dados I/Lesson #02")
```


Arquivos em Python

```
# Code from previous cells
f = open('crime_rates.csv', 'r')
data = f.read()
data
```

```
'Albuquerque,749\nAnaheim,371\nAnchorage,828\n.
```

"data" é uma grande string

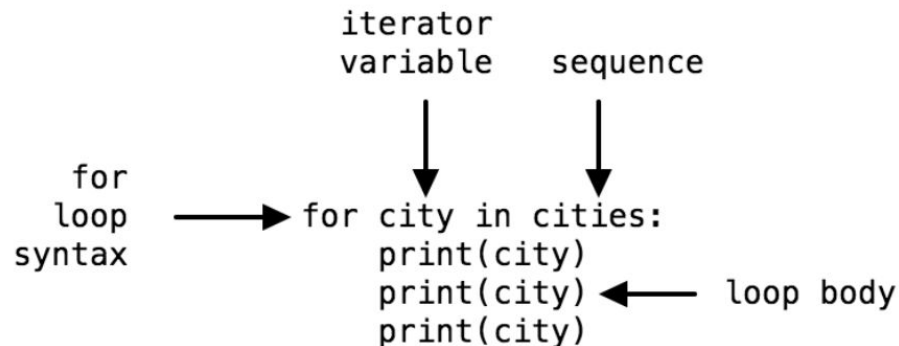
"data_list" é uma lista de strings

```
data_list = data.split("\n")
data_list
```

```
['Albuquerque,749',
 'Anaheim,371',
 'Anchorage,828',
 'Arlington,503',
 'Atlanta,1379',
 'Aurora,425',
```

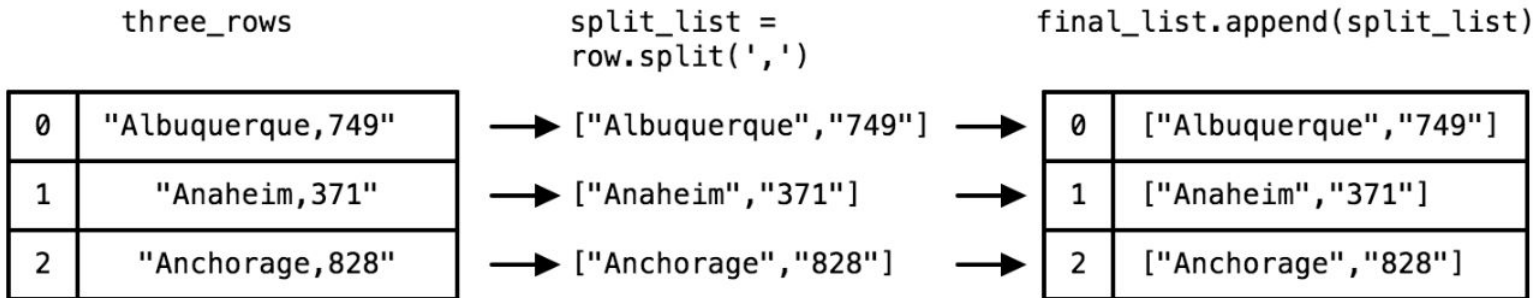
Estrutura de Repetição

```
for city in cities:  
    print(city)  
    print(city)  
    print(city)
```



Lista de Listas

```
three_rows = ["Albuquerque,749", "Anaheim,371", "Anchorage,828"]
final_list = []
for row in three_rows:
    split_list = row.split(',')
    final_list.append(split_list)
print(final_list)
```



Acessando Elementos de uma Lista de Listas

```
first_list = final_data[0] # Returns the first list: ['Albuquerque', '749'].
first_list_first_value = first_list[0] # Returns the first list's first element: 'Albuquerque'.
```

final_data
(list of lists)

0	['Albuquerque', '749']
1	['Anaheim', '371']
2	['Anchorage', '828']

final_data[0]
(list)

0	'Albuquerque'
1	'749'

final_data[0][0]
(string)

'Albuquerque'

Acessando Elementos de uma Lista de Listas

`final_data`
(list of lists)

0	['Albuquerque', '749']
1	['Anaheim', '371']
2	['Anchorage', '828']

`final_data[0]`
(list)

0	'Albuquerque'
1	'749'

`final_data[0][0]`
(string)

'Albuquerque'

```
crime_rates = []  
for row in five_elements:  
    crime_rate = row[1] # row is a list variable, not a string.  
    crime_rates.append(crime_rate) # crime_rate is a string, the crime rate of the city
```

Variáveis booleanas

```
t = True  
f = False
```

```
# True  
t = (8 == 8)  
# False  
u = (8 != 8)
```

```
"8" == "8"  
["January", "February"] == ["January", "February"]  
5.0 == 5.0
```

```
rates = [10, 15, 20]  
rates[0] > rates[1] # False  
rates[0] >= rates[0] # True
```

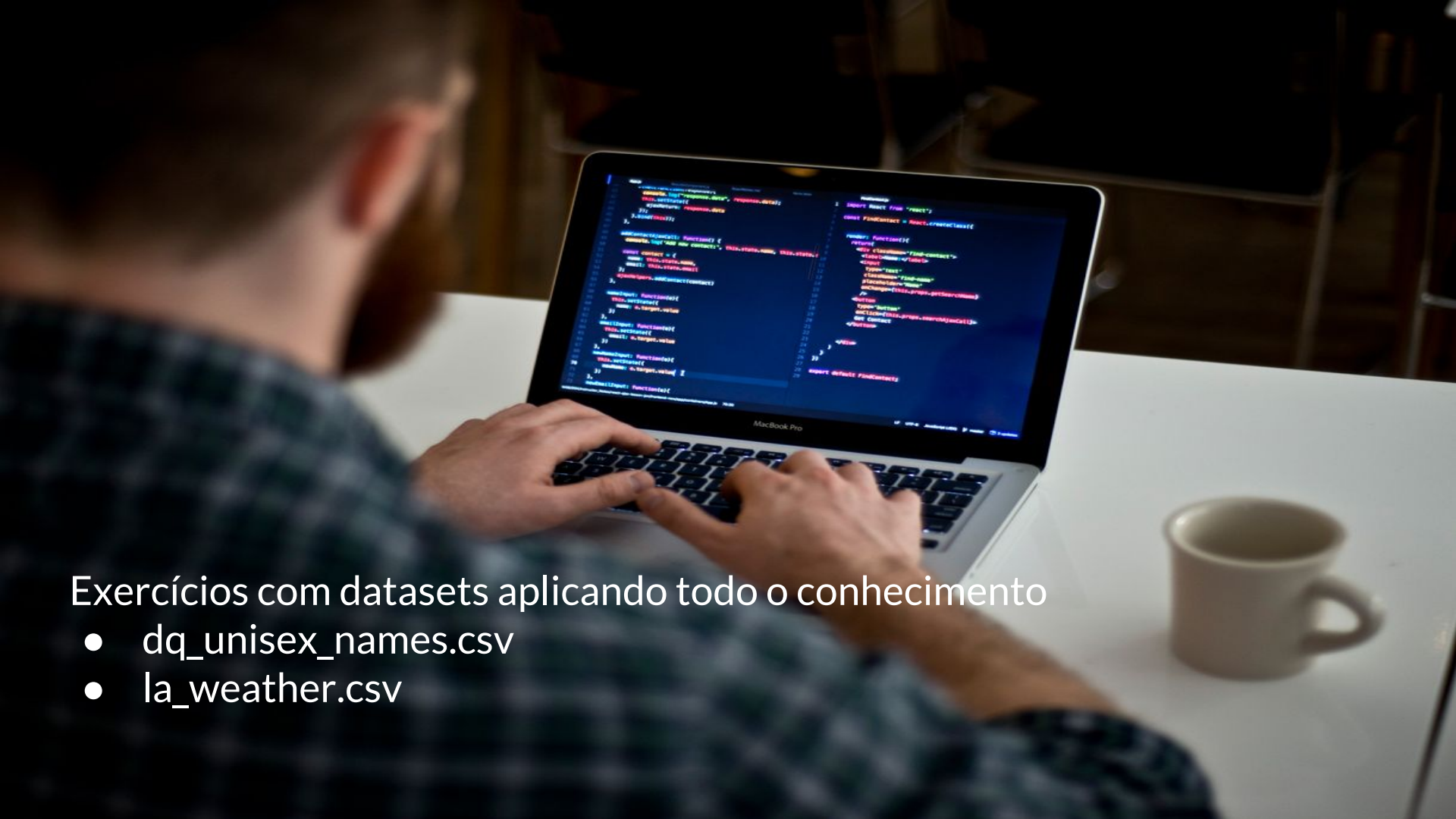
```
rates = [10, 15, 20]  
rates[0] < rates[1] # True  
rates[0] <= rates[0] # True
```

Estruturas condicionais

```
sample_rate = 749
greater = (sample_rate > 5)
if greater:
    print(sample_rate)
```

```
sample_rate = 749
greater = (sample_rate > 5)
if greater:
    print(sample_rate)
```

```
t = True
f = False
if t:
    print("Now you see me")
if f:
    print("Now you don't")
```



Exercícios com datasets aplicando todo o conhecimento

- dq_unisex_names.csv
- la_weather.csv