



UFCD 10794 - Programação avançada com Python

Notebook 09 - Introdução ao Pandas - Python

Sílvia Martins

AEFHP 2022

Biblioteca Pandas

A biblioteca Pandas é open-source e de uso gratuito, que fornece ferramentas para análise e manipulação de dados.

O pandas permite trabalhar com diferentes tipos de dados, por exemplo:

Dados em tabelas, como uma folha de cálculo Excel ou uma tabela MySQL; Dados ordenados; Arrays.

A magia de ler, manipular, agregar e exibir os dados com poucos comandos explica porque a biblioteca se tem tornado tão popular. As estruturas primárias do Pandas são: `Series` e `DataFrames`.

- Uma ***Serie*** é um array unidimensional (vetor) que contém uma sequência de valores que apresentam uma indexação (que podem ser numéricos inteiros ou rótulos), muito parecida com uma única coluna no Excel.
- Um ***DataFrame*** é uma estrutura de dados em tabela, semelhante a uma folha de cálculo de dados do Excel, em que tanto as linhas quanto as colunas apresentam rótulos.

A partir dos objetos principais, a biblioteca Pandas disponibiliza um conjunto de funcionalidades sofisticadas de indexação, que permite reformatar, manipular, agregar ou selecionar subconjuntos específicos dos dados em que se está a trabalhar.

Series

	nome		idade	
Índice	0	Maria	0	25
	1	José	1	56
	2	Ana	2	31
	3	Paulo	3	43
				Linhas

DataFrame

	Colunas	
	nomes	idade
0	Maria	25
1	José	56
2	Ana	31
3	Paulo	43

Destacam-se algumas formas de manusear um DataFrame:

- O método `head()` é um dos mais usados para obter uma visão geral rápida do DataFrame.
- O atributo `loc` é usado para devolver uma ou mais linhas especificadas
- O argumento `index` permite renomear os índices

Na plataforma de distribuição Anaconda, o Pandas já vem instalado por padrão. Caso não seja reconhecido deve-se proceder à sua instalação, no terminal, usando (ver ficha 1):

In []:

```
pip3 install pandas
```

Importação de ficheiros CSV com Pandas

Importar o Pandas e sua utilização para para leitura do ficheiro `empregados.csv`, criado anteriormente:

In [13]:

```
import pandas as pd
emp = pd.read_csv('empregados.csv')
print(emp)
```

	Nome	idade	cargo	departamento	salario
0	Ana	34	formador	A	1100
1	Rui	45	cozinheiro	C	900

Criação de ficheiros CSV com a Pandas

In [37]:

```

tartaruga={'name': ['Raphael', 'Donatello'],
           'mask': ['red', 'purple'],
           'weapon': ['sai', 'bo staff']}
df = pd.DataFrame(tartaruga) #criação de um DataFrame a partir de um dicionário
print(df)
print()
df.to_csv('tartarugas_ninja.csv', index=False)

emp = pd.read_csv('tartarugas_ninja.csv')
print(emp)
#'name,mask,weapon\nRaphael,red,sai\nDonatello,purple,bo staff\n'

```

	name	mask	weapon
0	Raphael	red	sai
1	Donatello	purple	bo staff

	name	mask	weapon
0	Raphael	red	sai
1	Donatello	purple	bo staff

Devolve a linha 0:

In [33]:

```

# faz referencia ao índice de linha
print(emp.loc[0])

```

```

name      Raphael
mask      red
weapon    sai
Name: 0, dtype: object

```

Devolve as linhas 0 e 1:

In [34]:

```

# Usa uma lista de índices
print(emp.loc[[0,1]])

```

	name	mask	weapon
0	Raphael	red	sai
1	Donatello	purple	bo staff

In [35]:

```

#devolve um Dataframe vazio
print(emp.loc[[]])

```

```

Empty DataFrame
Columns: [name, mask, weapon]
Index: []

```

In [79]:

```
df=pd.DataFrame(tartaruga,index=["Turtle1", "Turtle2"])
print(df)
print()
print(emp)
```

	name	mask	weapon
Turtle1	Raphael	red	sai
Turtle2	Donatello	purple	bo staff

	name	mask	weapon
0	Raphael	red	sai
1	Donatello	purple	bo staff

Exercício:

Cria um DataFrame, com o nome 'exercicio' que contenha as calorias e a duração de 3 exercicios específicos, sendo os valores das calorias gastas:420, 380 e 390 e a duração de 50, 40 e 45 minutos, respetivamente:

- Cria o Dataframe;
- Importa os dados;
- Exibe os dados do Dataframe;
- Exibe os dados, agora com os índices: Ex1, Ex2 e Ex3.
- Exibe o Dataframe, linha a linha.

Importação de ficheiros xml com Pandas

Importar o Pandas e sua utilização para para leitura do ficheiro *bookstore.xml*, ou exibição do mesmo convertido para string. bem como de outras string em formato XML:

In [57]:

```

xmlb = """<?xml version="1.0" encoding="UTF-8"?>
<bookstore>
  <book category="cooking">
    <title lang="en">Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>
  <book category="children">
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
  <book category="web">
    <title lang="en">Learning XML</title>
    <author>Erik T. Ray</author>
    <year>2003</year>
    <price>39.95</price>
  </book>
</bookstore>"""

xmlcat = """<?xml version="1.0" encoding="UTF-8"?>
<Animal>
  <GATO>
    <Nome>Felix</Nome>
    <RACA>Siames</RACA>
    <IDADE>6</IDADE>
    <LICENCA>IFF138bod</LICENCA>
    <DONO>Rodrigo Santoro</DONO>
  </GATO>
  <GATO>
    <Nome>Pandas</Nome>
    <RACA>Chartreux</RACA>
    <IDADE>4</IDADE>
    <LICENCA>Idd130bod</LICENCA>
    <DONO>Marta Temido</DONO>
  </GATO>
</Animal>"""

xml = '''<?xml version='1.0' encoding='utf-8'?>
<data>
  <row shape="square" degrees="360" sides="4.0"/>
  <row shape="circle" degrees="360"/>
  <row shape="triangle" degrees="180" sides="3.0"/>
</data>'''

```

In [73]:

```
import pandas as pd

df = pd.read_xml("bookstore.xml")
print("DO FICHEIRO:\n",df,"\n\n")
df = pd.read_xml(xml)
print("DA STRING:\n",df,"\n\n")
df = pd.read_xml(xmlcat)
print(df)
```

DO FICHEIRO:

	category	title	author	year	price
cover					
0	cooking	Everyday Italian	Giada De Laurentiis	2005	30.00
None					
1	children	Harry Potter	J K. Rowling	2005	29.99
None					
2	web	XQuery Kick Start	Vaidyanathan Nagarajan	2003	49.99
None					
3	web	Learning XML	Erik T. Ray	2003	39.95 paper
back					

DA STRING:

	category	title	author	year	price
0	cooking	Everyday Italian	Giada De Laurentiis	2005	30.00
1	children	Harry Potter	J K. Rowling	2005	29.99
2	web	Learning XML	Erik T. Ray	2003	39.95

Uma ferramenta poderosa é a capacidade de consultar nós de forma seletiva ou condicional com *XPath* :

In [63]:

```
df = pd.read_xml(xmlb, xpath="//book[year=2005]")
print(df)
```

	category	title	author	year	price
0	cooking	Everyday Italian	Giada De Laurentiis	2005	30.00
1	children	Harry Potter	J K. Rowling	2005	29.99

Especificar apenas elementos ou apenas atributos para análise:

In [74]:

```
df = pd.read_xml(xmlb, elems_only=True) # seleciona os elementos
print(df, "\n\n")
df = pd.read_xml(xmlb, attrs_only=True) # seleciona os atributos
print(df)
```

	title	author	year	price
0	Everyday Italian	Giada De Laurentiis	2005	30.00
1	Harry Potter	J K. Rowling	2005	29.99
2	Learning XML	Erik T. Ray	2003	39.95

	category
0	cooking
1	children
2	web

Bibliografia

<https://harve.com.br/blog/programacao-python-blog/pandas-python-vantagens-e-como-comecar/>
[\(https://harve.com.br/blog/programacao-python-blog/pandas-python-vantagens-e-como-comecar/\)](https://harve.com.br/blog/programacao-python-blog/pandas-python-vantagens-e-como-comecar/)
<https://pythonacademy.com.br/blog/python-e-virtualenv-como-programar-em-ambientes-virtuais>
<https://pythonacademy.com.br/blog/python-e-virtualenv-como-programar-em-ambientes-virtuais>
<https://pythonacademy.com.br/blog/importar-csv-no-pandas> <https://pythonacademy.com.br/blog/importar-csv-no-pandas>
https://pandas.pydata.org/docs/reference/api/pandas.read_xml.html
https://pandas.pydata.org/docs/reference/api/pandas.read_xml.html
https://pandas.pydata.org/docs/reference/api/pandas.read_excel.html
https://pandas.pydata.org/docs/reference/api/pandas.read_excel.html
https://pandas.pydata.org/docs/user_guide/io.html#io-read-xml
https://pandas.pydata.org/docs/user_guide/io.html#io-read-xml

