FUNDAMENTOS E TÉCNICAS EM CIÊNCIAS DE DADOS

PROF. JOSENALDE OLIVEIRA

josenalde@eaj.ufrn.br
https://github.com/josenalde/datascience

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - UFRN

PANDAS #2 – EXPLORANDO DATAFRAMES

- SELECIONAR COLUNAS, LINHAS, ITENS INDIVIDUAIS COM LOCALIZAÇÃO DE ÍNDICES
- MÉTODOS PANDAS PARA PRODUZIR ARRAYS BOOLEANOS
- OPERADORES BOOLEANOS COMBINADAS PARA REALIZAR OPERAÇÕES MAIS COMPLEXAS
- AGREGAÇÃO PARA ANÁLISE MAIS AVANÇADA COM LOOPS

```
d = np.random.randint(2,20, size=(30,5))
listIndex = []
for i in range(0,30):
    listIndex.append('i' + str(i))
df = pd.DataFrame(data=d, columns=list('ABCDE'), index=listIndex)
```

Rótulos

	A	B	C	D	E
i0	12	6	14	9	18
i1	16	13	3	4	13
i2	16	13	11	7	12
i3	18	3	5	11	17
i4	2	7	11	18	14
i5	17	2	17	18	15

df.loc['i2', 'D'] ?

df.loc['i5'] ?

df.loc[:, 'E'] ?

O iloc permite selecionar por índice inteiro

df.iloc[2, 3] ?

df.iloc[5] ?

df.iloc[:, 4] ?

PANDAS #2 – EXPLORANDO DATAFRAMES

- SELECIONAR COLUNAS, LINHAS, ITENS INDIVIDUAIS COM LOCALIZAÇÃO DE ÍNDICES
- MÉTODOS PANDAS PARA PRODUZIR ARRAYS BOOLEANOS
- OPERADORES BOOLEANOS COMBINADAS PARA REALIZAR OPERAÇÕES MAIS COMPLEXAS
- AGREGAÇÃO PARA ANÁLISE MAIS AVANÇADA COM LOOPS

df.iloc[1:3]

	A	B	C	D	E
i0	12	6	14	9	18
i1	16	13	3	4	13
i2	16	13	11	7	12
i3	18	3	5	11	17
i4	2	7	11	18	14
i5	17	2	17	18	15

Diferença: loc inclui extremo (fim)

df.loc['i1':'i3']

	A	B	C	D	E
i0	12	6	14	9	18
i1	16	13	3	4	13
i2	16	13	11	7	12
i3	18	3	5	11	17
i4	2	7	11	18	14
i5	17	2	17	18	15

PANDAS #2 – EXPLORANDO DATAFRAMES

- SELECIONAR COLUNAS, LINHAS, ITENS INDIVIDUAIS COM LOCALIZAÇÃO DE ÍNDICES
- MÉTODOS PANDAS PARA PRODUZIR ARRAYS BOOLEANOS
- OPERADORES BOOLEANOS COMBINADAS PARA REALIZAR OPERAÇÕES MAIS COMPLEXAS
- AGREGAÇÃO PARA ANÁLISE MAIS AVANÇADA COM LOOPS

df.iloc[1:3]

	A	B	C	D	E
i0	12	6	14	9	18
i1	16	13	3	4	13
i2	16	13	11	7	12
i3	18	3	5	11	17
i4	2	7	11	18	14
i5	17	2	17	18	15

Diferença: loc inclui extremo (fim)

df.loc['i1':'i3']

	A	B	C	D	E
i0	12	6	14	9	18
i1	16	13	3	4	13
i2	16	13	11	7	12
i3	18	3	5	11	17
i4	2	7	11	18	14
i5	17	2	17	18	15

PANDAS #2 – MÁSCARAS BOOLEANAS

Seja o dataset FORTUNE1000.csv

```
df2 = pd.read_csv('../datasets/fortune1000.csv')
```

	rank	title	Previous Rank	Revenues ($M)	Revenue Change	Profits ($M)	Profit Change	Assets ($M)	Value as of 3/29/18 ($M)	Employees	CEO	CEO Title	Sector	Industry	Years on Fortune 500 List	City	State	Latitude	Longitude
0	1	Walmart	1	$500,343	3.00%	$9,862.00	-27.70%	$204,522	$263,563	2,300,000	C. Douglas McMillon	President, Chief Executive Officer & Director	Retailing	General Merchandisers	24	Bentonville	AR	36.372854	-94.208817
1	2	Exxon Mobil	4	$244,363	17.40%	$19,710.00	151.40%	$348,691	$316,157	71,200	Darren W. Woods	Chairman & Chief Executive Officer	Energy	Petroleum Refining	24	Irving	TX	32.814018	-96.948894
2	3	Berkshire Hathaway	2	$242,137	8.30%	$44,940.00	86.70%	$702,095	$492,008	377,000	Warren E. Buffett	Chairman, President & Chief Executive Officer	Financials	Insurance: Property and Casualty (Stock)	24	Omaha	NE	41.256537	-95.934503
3	4	Apple	3	$229,234	6.30%	$48,351.00	5.80%	$375,319	$851,318	123,000	Timothy D. Cook	Chairman & Chief Executive Officer	Technology	Computers, Office Equipment	24	Cupertino	CA	37.322998	-122.032182
4	5	UnitedHealth Group	6	$201,159	8.80%	$10,558.00	50.50%	$139,058	$207,080	260,000	David S. Wichmann	Chairman & Chief Executive Officer	Health Care	Health Care: Insurance and Managed Care	24	Minnetonka	MN	44.921184	-93.468749

```
df2.columns
```

```
Index(['rank', 'title', 'Previous Rank', 'Revenues ($M)',
'Revenue Change', 'Profits ($M)', 'Profit Change', 'Assets ($M)', 'Mkt Value as of 3/29/18 ($M)',
'Employees', 'CEO', 'CEO Title', 'Sector', 'Industry', 'Years on Fortune 500 List', 'City', 'State',
'Latitude', 'Longitude'], dtype='object')
```

PANDAS #2 – MÁSCARAS BOOLEANAS

```
Index(['rank', 'title', 'Previous Rank', 'Revenues ($M)',
'Revenue Change', 'Profits ($M)', 'Profit Change', 'Assets ($M)', 'Mkt Value as of 3/29/18 ($M)',
'Employees', 'CEO', 'CEO Title', 'Sector', 'Industry', 'Years on Fortune 500 List', 'City', 'State',
'Latitude', 'Longitude'], dtype='object')
```

```
is_computer = df2['Industry'].str.startswith('Computers') #cria a mascara booleana
is_computer
```

```
0      False
1      False
2      False
3       True
4      False
       ...
995    False
996    False
997    False
998    False
999    False
Name: Industry, Length: 10
```

	rank	title	Previous Rank	Revenues ($M)	Revenue Change	Profits ($M)	Profit Change	Assets ($M)	Value as of 3/29/18 ($M)	Employees	CEO	CEO Title	Sector	Industry	Fo 50
0	1	Walmart	1	$500,343	3.00%	$9,862.00	-27.70%	$204,522	$263,563	2,300,000	C. Douglas McMillon	President, Chief Executive Officer & Director	Retailing	General Merchandisers	
1	2	Exxon Mobil	4	$244,363	17.40%	$19,710.00	151.40%	$348,691	$316,157	71,200	Darren W. Woods	Chairman & Chief Executive Officer	Energy	Petroleum Refining	
2	3	Berkshire Hathaway	2	$242,137	8.30%	$44,940.00	86.70%	$702,095	$492,008	377,000	Warren E. Buffett	Chairman, President & Chief Executive Officer	Financials	Insurance: Property and Casualty (Stock)	
3	4	Apple	3	$229,234	6.30%	$48,351.00	5.80%	$375,319	$851,318	123,000	Timothy D. Cook	Chairman & Chief Executive Officer	Technology	Computers, Office Equipment	
4	5	UnitedHealth Group	6	$201,159	8.80%	$10,558.00	50.50%	$139,058	$207,080	260,000	David S. Wichmann	Chairman & Chief Executive Officer	Health Care	Health Care: Insurance and Managed Care	

```
cols = ['rank', 'title', 'Profits ($M)', 'Employees']
df2.loc[is_computer, cols] #aplica a máscara nas colunas
```

PANDAS #2 – MÁSCARAS BOOLEANAS

```
Index(['rank', 'title', 'Previous Rank', 'Revenues ($M)',
'Revenue Change', 'Profits ($M)', 'Profit Change', 'Assets ($M)', 'Mkt Value as of 3/29/18 ($M)',
'Employees', 'CEO', 'CEO Title', 'Sector', 'Industry', 'Years on Fortune 500 List', 'City', 'State',
'Latitude', 'Longitude'], dtype='object')
```

```
is_computer = df2['Industry'].str.startswith('Computers') #cria a mascara booleana
is_computer
```

```
cols = ['rank', 'title', 'Profits ($M)', 'Employees']
df2.loc[is_computer, cols] #aplica a máscara nas colunas
```

	rank	title	Profits ($M)	Employees
3	4	Apple	$48,351.00	123,000
34	35	Dell Technologies	($3,728.00)	145,000
57	58	HP	$2,526.00	49,000
106	107	Hewlett Packard Enterprise	$344.00	66,000
157	158	Western Digital	$397.00	67,629
290	291	Xerox	$195.00	36,100
431	432	NCR	$232.00	34,000
494	495	NetApp	$509.00	10,100
550	551	Diebold Nixdorf	($233.10)	23,000
656	657	Pitney Bowes	$261.30	14,700
823	824	Super Micro Computer	$69.30	2,699

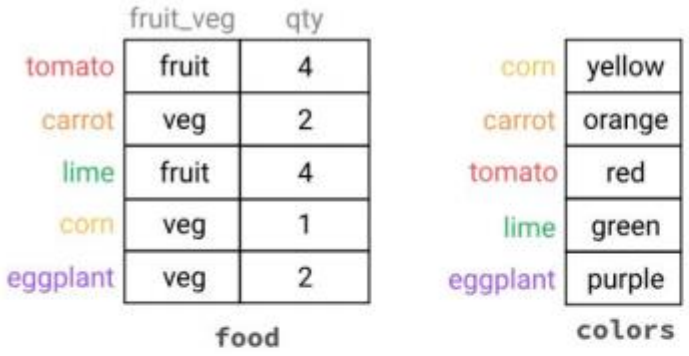
Coluna Employees com
strings com separadores vírgula
Função de conversão str-float
Defeituosa neste caso! Remover virgulas

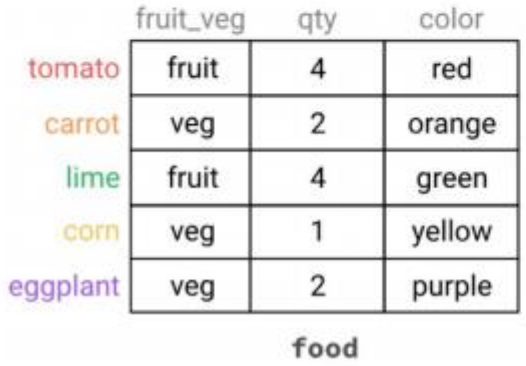
```
print('------- antes --------')
print(df2.dtypes)
# Comparação incorreta, pois o campo Employees é object-string
#c1 = df2['Employees'] > 300000
df2['Employees'] = df2['Employees'].str.replace(',', '')
print(df2)
print('------- depois --------')
df2['Employees'] = df2['Employees'].astype(float)
print(df2.dtypes)
c1 = df2['Employees'] > 70000
c2 = df2['Sector'] == 'Energy'
cond = c1 & c2 #máscara
cols = ['rank', 'title', 'Sector', 'Employees']
df2.loc[cond, cols]
```

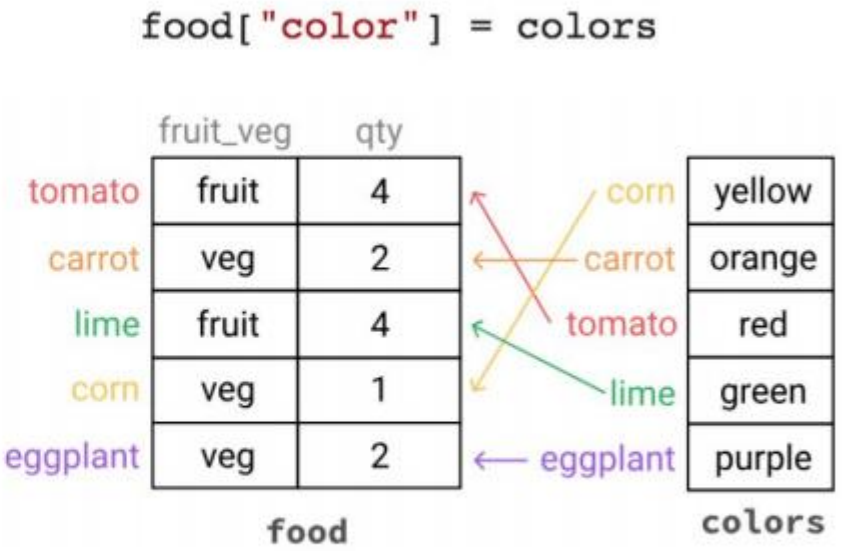
	rank	title	Sector	Employees
1	2	Exxon Mobil	Energy	71200.0

PANDAS #2 – ALINHAMENTO DE ÍNDICES

```
Criar dataframes e comandos que reproduzam este exemplo:
```







https://pandas.pydata.org/docs/reference/api/pandas.concat.html?highlight=concat

PANDAS #2 – ALINHAMENTO DE ÍNDICES

Operações aritméticas entre dataframes e series

```
O r é o método com argumento
invertido.
Exemplo

1/df

df.rdiv(1)
```

Método	Descrição
`add`, `radd`	Métodos para adição (+)
`sub`, `rsub`	Métodos para subtração (-)
`div`, `rdiv`	Métodos para divisão (/)
`floordiv`, `rfloordiv`	Métodos para divisão pelo piso (//)
`mul`, `rmul`	Métodos para multiplicação (*)
`pow`, `rpow`	Métodos para exponencial (**)

PANDAS #2 – PARA IR ALÉM

```
Estudar o notebook de exemplo com análise dos commits do Kernel Linux ao longo da história
```

Exploring-the-evolution-of-Linux-Datacamp-Project/notebook.ipynb at master · vneogi199/Exploring-the-evolution-of-Linux-Datacamp-Project · GitHub