THE DEVELOPER'S CONFERENCE

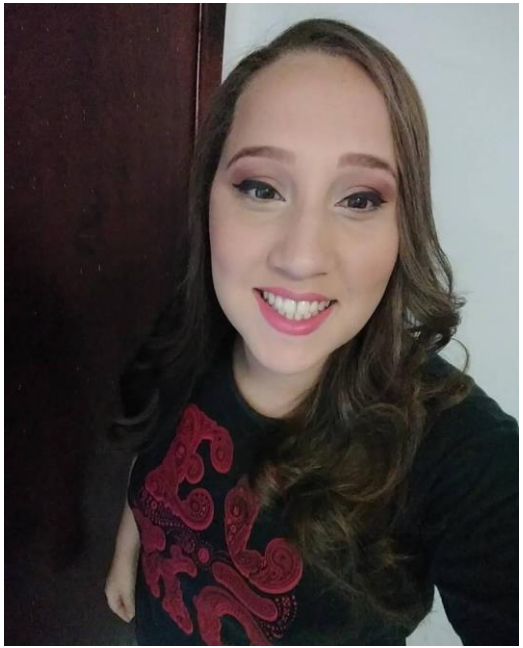
Python Para análise de dados

Fernanda Santos
Gestora da Informação

Apresentação

THE DEVELOPER'S CONFERENCE

Pernambucana
Gestora Da Informação(UFPE)
Analista de dados(Itaú)
Apaixonada por transformar dados em informação



Pandas

THE DEVELOPER'S CONFERENCE

- Pandas é uma biblioteca Python de código aberto para análise de dados. Ele dá ao Python a capacidade de trabalhar com dados tipo planilha, permitindo carregar, manipular, alinhar e combinar dados rapidamente, entre outras funções.

Conjunto de dados csv

THE DEVELOPER'S CONFERENCE

www.github.com/jennybc/gapminder

Carregando o Dataset

THE DEVELOPER'S CONFERENCE

```
In [38]: import pandas as pd
```

```
In [39]: df = pd.read_csv('dados/gapminder.tsv', sep='\t')
```

Visualizando as primeiras linhas

THE DEVELOPER'S CONFERENCE

```
In [40]: df.head()
```

Out[40]:

	country	continent	year	lifeExp	pop	gdpPercap
0	Afghanistan	Asia	1952	28.801	8425333	779.445314
1	Afghanistan	Asia	1957	30.332	9240934	820.853030
2	Afghanistan	Asia	1962	31.997	10267083	853.100710
3	Afghanistan	Asia	1967	34.020	11537966	836.197138
4	Afghanistan	Asia	1972	36.088	13079460	739.981106

THE DEVELOPER'S CONFERENCE

Tem como saber apenas a quantidade de linhas e colunas?

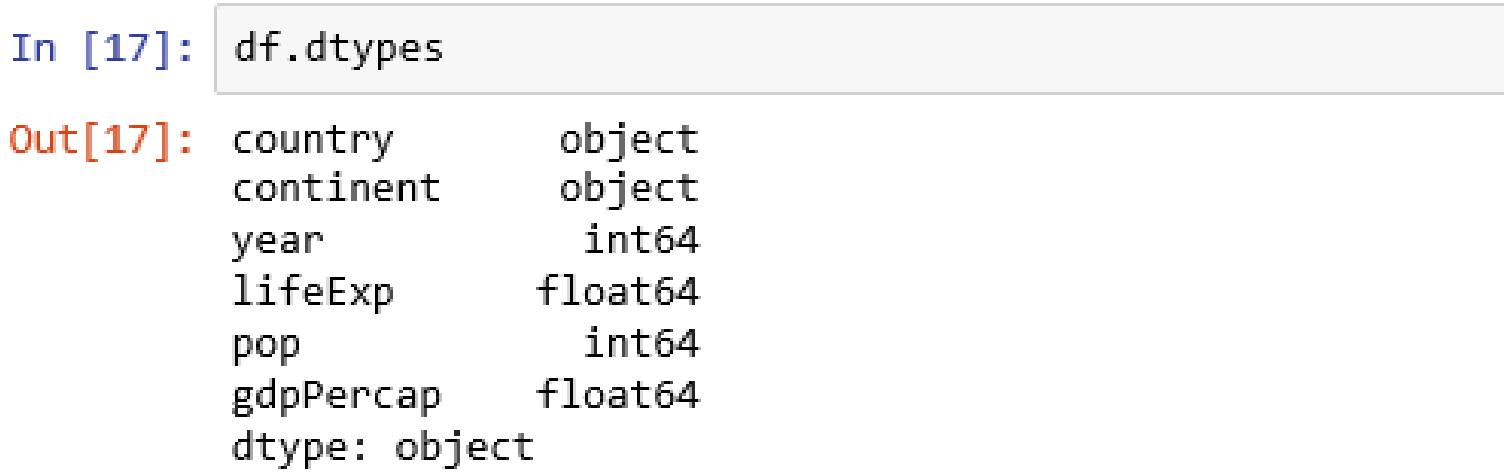
```
In [12]: df.shape
Out[12]: (1704, 6)
```

Retornando apenas os nomes das colunas

THE DEVELOPER'S CONFERENCE

```
In [14]: df.columns
Out[14]: Index(['country', 'continent', 'year', 'lifeExp', 'pop', 'gdpPercap'], dtype='object')
```

E como sei o tipo de informação que contém em cada coluna?



Método para exibir informações estatísticas

THE DEVELOPER'S CONFERENCE

```
In [72]: df.describe()
```

Out[72]:

	year	lifeExp	pop	gdpPercap
count	1704.00000	1704.000000	1.704000e+03	1704.000000
mean	1979.50000	59.474439	2.960121e+07	7215.327081
std	17.26533	12.917107	1.061579e+08	9857.454543
min	1952.00000	23.599000	6.001100e+04	241.165877
25%	1965.75000	48.198000	2.793664e+06	1202.060309
50%	1979.50000	60.712500	7.023596e+06	3531.846989
75%	1993.25000	70.845500	1.958522e+07	9325.462346
max	2007.00000	82.603000	1.318683e+09	113523.132900

Método Loc

THE DEVELOPER'S CONFERENCE

```
In [73]: oceania = df.loc[df["continent"] == "Oceania"]
         oceania.head(15)
```

Out[73]:

	country	continent	year	lifeExp	pop	gdpPercap
60	Australia	Oceania	1952	69.120	8691212	10039.59564
61	Australia	Oceania	1957	70.330	9712569	10949.64959
62	Australia	Oceania	1962	70.930	10794968	12217.22686
63	Australia	Oceania	1967	71.100	11872264	14526.12465
64	Australia	Oceania	1972	71.930	13177000	16788.62948
65	Australia	Oceania	1977	73.490	14074100	18334.19751
66	Australia	Oceania	1982	74.740	15184200	19477.00928
67	Australia	Oceania	1987	76.320	16257249	21888.88903
68	Australia	Oceania	1992	77.560	17481977	23424.76683
69	Australia	Oceania	1997	78.830	18565243	26997.93657
70	Australia	Oceania	2002	80.370	19546792	30687.75473
71	Australia	Oceania	2007	81.235	20434176	34435.36744
1092	New Zealand	Oceania	1952	69.390	1994794	10556.57566
1093	New Zealand	Oceania	1957	70.260	2229407	12247.39532
1094	New Zealand	Oceania	1962	71.240	2488550	13175.67800

Para cada ano em nossos dados, qual era a expectativa de vida média?

THE DEVELOPER'S CONFERENCE

```
In [53]: df.groupby('year')['lifeExp'].mean()

Out[53]: year
         1952    49.057620
         1957    51.507401
         1962    53.609249
         1967    55.678290
         1972    57.647386
         1977    59.570157
         1982    61.533197
         1987    63.212613
         1992    64.160338
         1997    65.014676
         2002    65.694923
         2007    67.007423
         Name: lifeExp, dtype: float64
```

Trabalhando com Excel

THE DEVELOPER'S CONFERENCE

```
In [58]: import pandas as pd
```

```
In [65]: df = pandas.read_excel('dados/planilha_vendas.xlsx')
```

```
In [66]: df.head()
```

Out[66]:

	Cidade ID	Data	Vendas	LojaID	Qtde
0	22.0	2014-01-02	127.92	1036.0	1
1	22.0	2014-01-02	30.97	1036.0	1
2	22.0	2014-01-02	39.29	1036.0	1
3	22.0	2014-01-02	158.66	1036.0	1
4	22.0	2014-01-02	242.31	1036.0	1

Alterando o tipo da coluna

THE DEVELOPER'S CONFERENCE

```
In [72]: df["LojaID"] = df["LojaID"].astype("object")
```

```
In [73]: df.head()
```

Out[73]:

	Cidade ID	Data	Vendas	LojaID	Qtde
0	22.0	2014-01-02	127.92	1036	1
1	22.0	2014-01-02	30.97	1036	1
2	22.0	2014-01-02	39.29	1036	1
3	22.0	2014-01-02	158.66	1036	1
4	22.0	2014-01-02	242.31	1036	1

agora já temos LojaID como string, como temos certeza disso? basta utilizar o df.dtypes como mostrado abaixo:

```
In [74]: df.dtypes

Out[74]: Cidade ID            float64
         Data          datetime64[ns]
         Vendas               float64
         LojaID                object
         Qtde                   int64
         dtype: object
```

Criando uma coluna para descobrir a Receita

THE DEVELOPER'S CONFERENCE

```
In [91]: df["Receita"] = df["Vendas"].mul(df["Qtde"])
```

```
In [92]: df.tail()
```

Out[92]:

	Cidade ID	Data	Vendas	LojaID	Qtde	Receita
507	26.0	2014-01-01	15.62	1037	2	31.24
508	26.0	2014-01-01	13.41	1037	7	93.87
509	26.0	2014-01-01	33.12	1037	9	298.08
510	26.0	2014-01-01	37.49	1037	2	74.98
511	26.0	2014-01-01	13.70	1037	6	82.20

Tratando Valores Faltantes

THE DEVELOPER'S CONFERENCE

```
In [80]: #Consultando linhas com valores faltantes
         df.isnull().sum()
```

```
Out[80]: Cidade ID    2
         Data         0
         Vendas       0
         LojaID       0
         Qtde         0
         Receita      0
         dtype: int64
```

Podemos apagar as linhas com valores nulos?

THE DEVELOPER'S CONFERENCE

```
In [88]: df.dropna(inplace=True)

In [89]: df.isnull().sum()

Out[89]: Cidade ID    0
         Data         0
         Vendas       0
         LojaID       0
         Qtde         0
         Receita      0
         dtype: int64
```

Contagem de Valores

THE DEVELOPER'S CONFERENCE

```
In [31]: df["LojaID"].value_counts()

Out[31]: 1037.0    398
         1036.0    103
         1035.0      7
         Name: LojaID, dtype: int64
```

Tem como visualizar em um gráfico?

THE DEVELOPER'S CONFERENCE

```
In [44]: %matplotlib inline
         df["LojaID"].value_counts().plot.bar()

Out[44]: <matplotlib.axes._subplots.AxesSubplot at 0x184b2fe46a0>
```

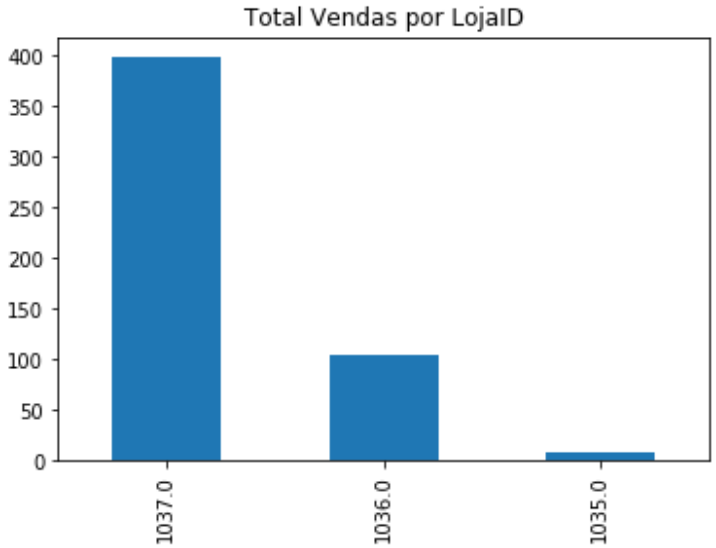
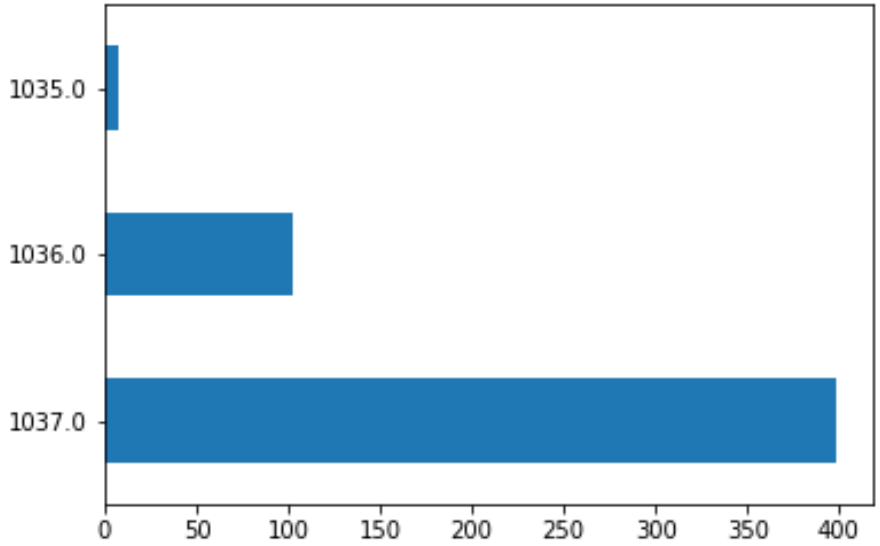


Gráfico de barras horizontais

THE DEVELOPER'S CONFERENCE

```
In [45]: %matplotlib inline
         df["LojaID"].value_counts().plot.barh()
```

```
Out[45]: <matplotlib.axes._subplots.AxesSubplot at 0x184b2fc2f60>
```



```
In [46]: %matplotlib inline
         df["LojaID"].value_counts(ascending=True).plot.barh()
```

```
Out[46]: <matplotlib.axes._subplots.AxesSubplot at 0x184b303a2e8>
```

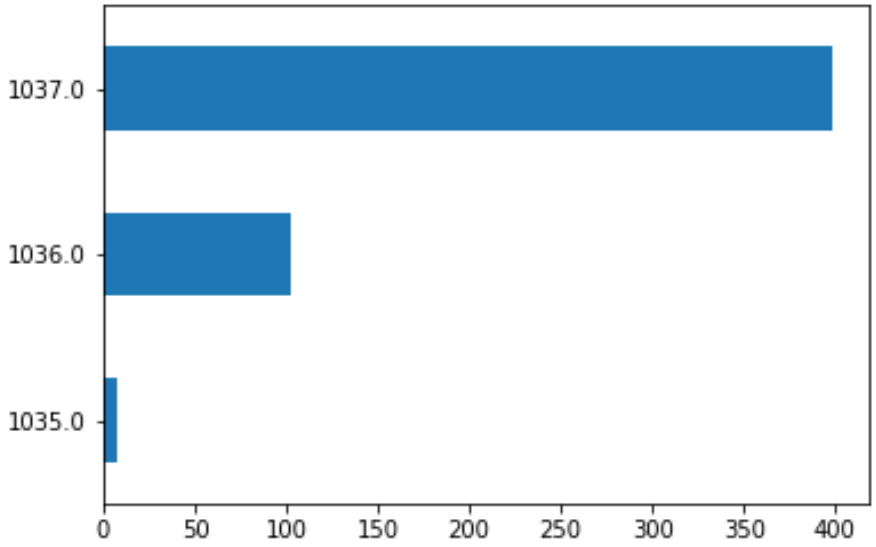
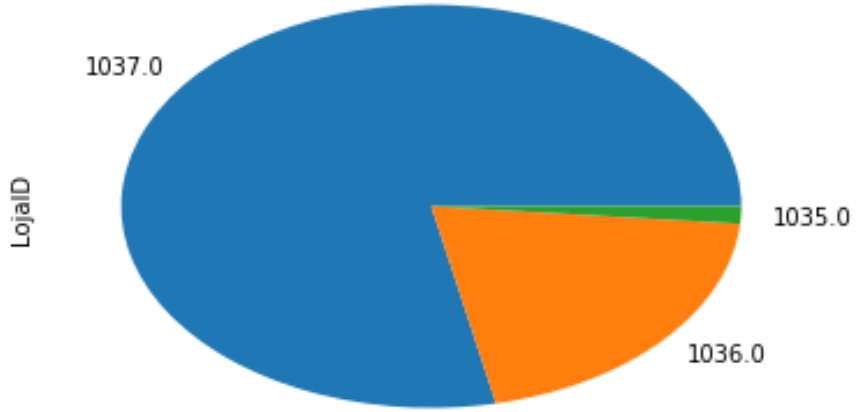


Gráfico de Pizza

THE DEVELOPER'S CONFERENCE

```
In [47]: %matplotlib inline
         df["LojaID"].value_counts().plot.pie()
```

```
Out[47]: <matplotlib.axes._subplots.AxesSubplot at 0x184b2f5b630>
```

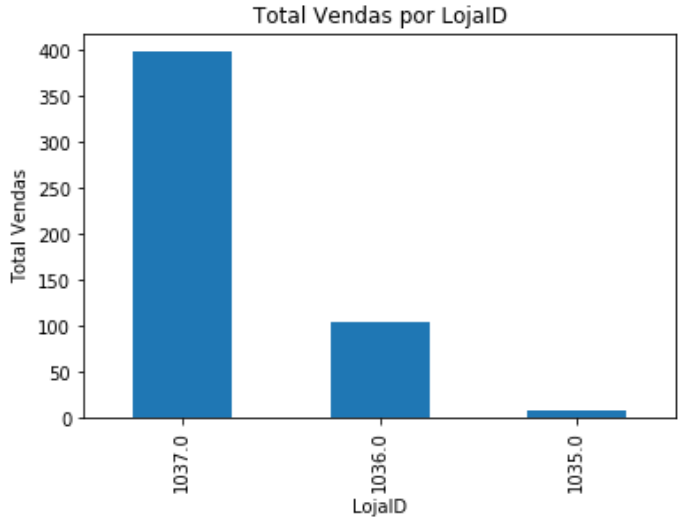


Adicionando um título e alterando o nome dos eixos

THE DEVELOPER'S CONFERENCE

```
In [50]: %matplotlib inline
         df["LojaID"].value_counts().plot.bar(title="Total Vendas por LojaID")
         plt.xlabel("LojaID")
         plt.ylabel("Total Vendas")
```

```
Out[50]: Text(0,0.5,'Total Vendas')
```



THE DEVELOPER'S CONFERENCE

Contatos

THE DEVELOPER'S CONFERENCE

- Linkedin: https://www.linkedin.com/in/fernanda-santos-18a821103
- Instagram: @ftspublicidade
- E-mail: ftspublicidade@gmail.com
- Github: github.com/Ftsnba

THE DEVELOPER'S CONFERENCE

OBRIGADA!!!!