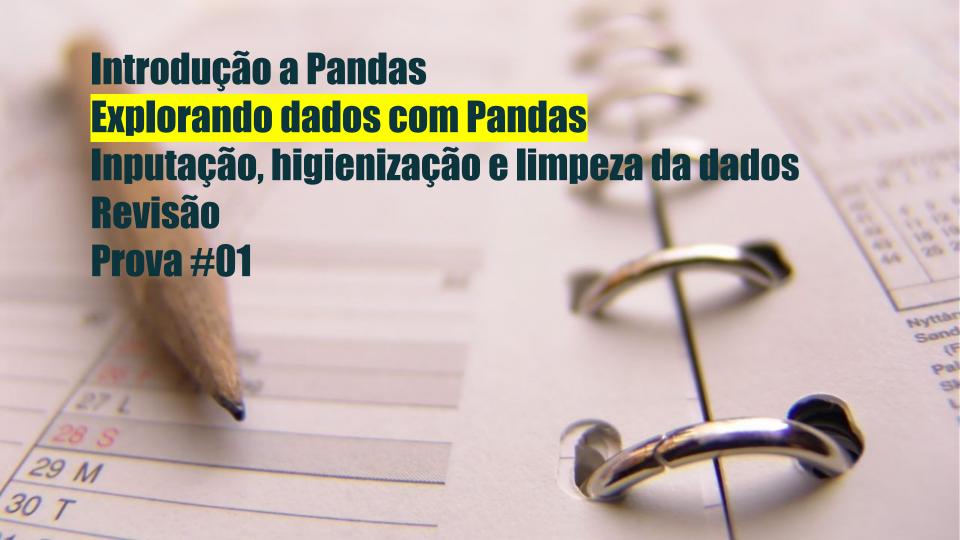




IMD033 - Probabilidade Aula 06 - Exploração de Dados com Pandas

Ivanovitch Silva Março 2019





Atualizar o repositório

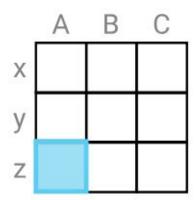
git clone https://github.com/ivanovitchm/imd0033_2019_1

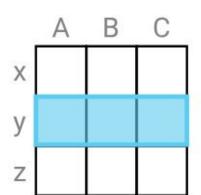
Ou

git pull











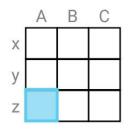
located at row with label z, column with label A

df.loc["y"]

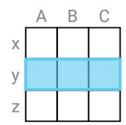
located at row with label y



Using iloc to select by integer position



df.iloc[2,0]



df.iloc[1]

```
first column = f500.iloc[:,0]
print(first column)
                          Walmart
                       State Grid
                   Sinopec Group
497
       Wm. Morrison Supermarkets
498
                              TUI
499
                       AutoNation
Name: company, dtype: object
```



Slicing with iloc

With **loc**[], the ending slice is **included**. With iloc[], the ending slice is not included.

Petroleum

1 f500[1:4]

	rank	revenues
State Grid	2	315199
Sinopec Group	3	267518
China National	4	262573

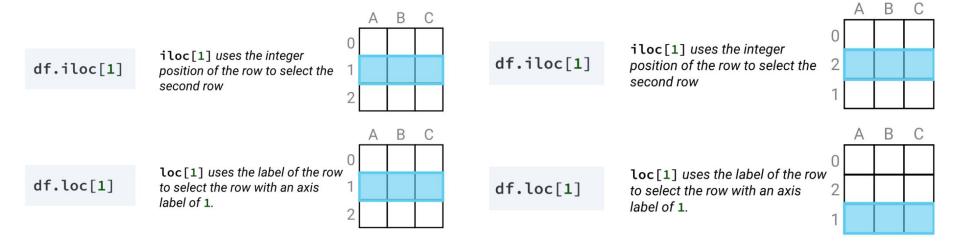
1 f500.iloc[1:4]

	rank	revenues
State Grid	2	315199
Sinopec Group	3	267518
China National Petroleum	4	262573





Loc vs iLoc





Using pandas methods to create boolean masks

```
>>> is california = usa["hq location"].str.endswith("CA")
>>> print(is california.head())
                                                      Bentonville, AR
                                               0
           False
    0
                                                            Omaha, NE
           False
                                                        Cupertino, CA
                                               8
                                               9
                                                           Irving, TX
    8
             True
                                               10
                                                     San Francisco, CA
           False
                                               Name: hg location, dtype: object
    10
            True
    Name: hq location, dtype: bool
```



Using boolean operators to select items

	company	revenues	country
0	Walmart	485873	USA
1	State Grid	315199	China
2	Sinopec Group	267518	China
3	China Nation	262573	China
4	Toyota Motor	254694	Japan

f500_sel

over_265 = f500_sel["revenues"] > 265000 china = f500_sel["country"] == "China"

True True True 3 False False over 265

False True True True False china

combined = over_265 & china

```
False
   True
                               False
                        =
   True
                True
                                True
                        =
   True
          &
                True
                               True
                        =
  False
                               False
                True
  False
          & 4
               False
                               False
               china
                             combined
over_265
```





Using boolean operators to select items

```
final_cols = ["company","revenues"]
result = f500_sel.loc[combined,final_cols]
                                        country
                  company
                              revenues
                Walmart
   False
                               485873
                                       USA
                                                        company
                                                                    revenues
                State Grid
                               315199
                                       China
    True
                                                      State Grid
                                                                     315199
                Sinopec Group
                               267518
                                       China
   True
                                                      Sinopec Group
                                                                     267518
                                       China
   False
                China Nation...
                               262573
                                                              result
   False
                Toyota Motor
                               254694
                                       Japan
 combined
                          f500 sel
```



Pandas Index Alignment

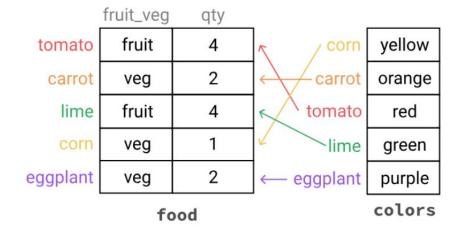
12	fruit_veg	qty
tomato	fruit	4
carrot	veg	2
lime	fruit	4
corn	veg	1
eggplant	veg	2

food

corn yellow
carrot orange
tomato red
lime green
eggplant purple
colors

fruit_veg qty color tomato fruit 4 red 2 carrot veg orange fruit lime 4 green yellow veg eggplant 2 purple veg

food["color"] = colors







Pandas Index Alignment

eggplant aubergine corn maize

alt_name

food["alt_name"] = alt_name fruit_veg color alt_name qty fruit NaN tomato 4 red 2 NaN carrot veg orange lime fruit 4 NaN green yellow maize corn veg eggplant 2 purple aubergine veg

food

Using Loops in Pandas

```
>>> print(df)
          В
>>> for i in df:
        print(i)
    A
    В
```

```
Because one of the key benefits of pandas is
that it has vectorized methods to work with
data more efficiently, we want to avoid using
loops wherever we can
```



Challenge: calculating return on assets by sector

```
{'Aerospace & Defense': 'Lockheed Martin',
 'Apparel': 'Nike',
 'Business Services': 'Adecco Group',
 'Chemicals': 'LyondellBasell Industries',
 'Energy': 'National Grid',
 'Engineering & Construction': 'Pacific Construction Group',
 'Financials': 'Berkshire Hathaway',
 'Food & Drug Stores': 'Publix Super Markets',
 'Food, Beverages & Tobacco': 'Philip Morris International',
 'Health Care': 'Gilead Sciences',
 'Hotels, Restaurants & Leisure': 'McDonald\xe2\x80\x99s',
 'Household Products': 'Unilever',
 'Industrials': '3M',
 'Materials': 'CRH',
 'Media': 'Disney',
                                                               return on assets = \frac{profits}{}
 'Motor Vehicles & Parts': 'Subaru',
 'Retailing': 'H & M Hennes & Mauritz',
 'Technology': 'Accenture',
 'Telecommunications': 'KDDI',
 'Transportation': 'Delta Air Lines',
```

'Wholesalers': 'McKesson'}



Lesson#06 Exploring Data with pandas.ipynb