Web Scraping com Beautiful Soup

Utilizaremos o Beautiful Soup do pacote BS4 para extrair a média salarial da profissão cientista de dados do site Glasdoor

In [169…

```
import requests
from bs4 import BeautifulSoup
import re
import pandas as pd
```

In [ ]:

```
headers = {'user-agent': 'Mozilla/5.0'} # evitar ser detectado como um bot
response = requests.get(
    'https://www.glassdoor.com.br/Sal%C3%A1rios/cientista-de-dados-sal%C3%A1rio-
    headers = headers
)

#response.text
```

In [ ]:

```
soup = response.text
clear_soup = BeautifulSoup(soup, "html.parser")
```

Buscar dados apenas da tag h3

In [57]:

```
list_of_cia = clear_soup.find_all("h3", {"data-test":"salaries-list-item-0-emplo
```

In [58]:

```
list_of_cia[0]
```

Out[58]:

```
<h3 class="m-0 css-g261rn" data-test="salaries-list-item-0-employer-name"><styl
e data-emotion-css="f3vw95">.css-f3vw95{cursor:pointer;font-size:15px;line-heig
ht:24px;color:#1861bf;font-size:inherit;}.css-f3vw95:hover{color:#0c4085;}</sty
le><a class="css-f3vw95 e1aj7ssy3" href="/Salário/Itaú-Unibanco-Itaú-BBA-e-Rede
-Cientista-De-Dados-Salários-E10999_D_KO30,48.htm?filter.payPeriod=MONTHLY">Ita
ú Unibanco (Itaú BBA e Rede)</a></h3>
```

In [26]:

```
list_of_cia[0].contents[1]
```

Out[26]:

```
<a class="css-f3vw95 e1aj7ssy3" href="/Salário/Itaú-Unibanco-Itaú-BBA-e-Rede-Ci
entista-De-Dados-Salários-E10999_D_KO30,48.htm?filter.payPeriod=MONTHLY">Itaú U
nibanco (Itaú BBA e Rede)</a>
```

In [31]:

```
list_of_cia[0].contents[0]
```

Out[31]:

```
<style data-emotion-css="f3vw95">.css-f3vw95{cursor:pointer;font-size:15px;line
-height:24px;color:#1861bf;font-size:inherit;}.css-f3vw95:hover{color:#0c4085;}
</style>
```

In [27]:

```
a = list_of_cia[0].contents[1].text
a
```

Out[27]:

```
'Itaú Unibanco (Itaú BBA e Rede)'
```

In [62]:

```
list_of_cia = clear_soup.find_all("h3", {"data-test":re.compile("salaries-list-i
```

In [63]:
```
len(list_of_cia)
```

Out[63]:
```
20
```

Listar as empresas empregadoras

In [64]:
```
for i in list_of_cia:
    print(i.find("a").text)
```

```
Itaú Unibanco (Itaú BBA e Rede)
IBM
Semantix
Hospital Israelita Albert Einstein
Banco Bradesco
Propz
Radix Engenharia e Software
TOTVS
Stefanini
Softplan
Autônomo (Brazil)
Grupo Globo
Globo
Ambev Tech
Ambev
Dasa
Nubank
Via
Aquarela Advanced Analytics
Banco do Brasil
```

In [115…
```
salary = clear_soup.find_all("div", {"data-test":re.compile(".*[0-9]-salary-info
```

In [116…
```
len(salary)
```

Out[116…
```
20
```

In [177…
```
for i in salary:
    s = i.find("h3").text
    print(re.sub(r"[R$\s]", "", s))
```

```
8.098
5.725
8.517
12.869
6.786
7.170
8.139
11.490
7.025
10.566
5.242
8.636
10.396
9.740
8.714
8.216
12.471
10.483
5.000
6.743
```

Criar o data frame

In [182…

```python
# lista com o nome das empresas
cia = []

for i in list_of_cia:
    cia.append(i.find("a").text)

# Criar lista com os salários pagos por cada empresa
sal = []

for i in salary:
    s = (i.find("h3").text)
    sal.append(re.sub(r"[R$\s]", "", s))
```

In [190…

```python
# criar o data frame
df = pd.DataFrame({'Empresa': cia,
                 'Salário': sal})

# Salvar como .csv
df.to_csv('salario_ds.csv',
          index=False,
          encoding='latin1',
          sep=';',
         decimal=',')
```

In [ ]:

```python

```