Web Scraping Python Code

February 11, 2020

0.1 Web Scraping_Sample Code

Shih-Hung (Angela) Ma

[1]:
```python
from bs4 import BeautifulSoup
from urllib.request import urlopen
import pandas as pd
```

[2]:
```python
####List of Presidents:

html = urlopen("https://en.wikipedia.org/wiki/
 ↪List_of_presidents_of_the_United_States")
soup = BeautifulSoup(html)

table = soup.find("table", {"class":"wikitable"})
table

for link in table.find_all("b"):
    name = link.find("a")
    print(name.get_text())

for link in table.find_all("b"):
    print(link.find("a").get_text())
```

```
George Washington
John Adams
Thomas Jefferson
James Madison
James Monroe
John Quincy Adams
Andrew Jackson
Martin Van Buren
William Henry Harrison
John Tyler
James K. Polk
Zachary Taylor
Millard Fillmore
Franklin Pierce
James Buchanan
```

Abraham Lincoln
Andrew Johnson
Ulysses S. Grant
Rutherford B. Hayes
James A. Garfield
Chester A. Arthur
Grover Cleveland
Benjamin Harrison
Grover Cleveland
William McKinley
Theodore Roosevelt
William Howard Taft
Woodrow Wilson
Warren G. Harding
Calvin Coolidge
Herbert Hoover
Franklin D. Roosevelt
Harry S. Truman
Dwight D. Eisenhower
John F. Kennedy
Lyndon B. Johnson
Richard Nixon
Gerald Ford
Jimmy Carter
Ronald Reagan
George H. W. Bush
Bill Clinton
George W. Bush
Barack Obama
Donald Trump
George Washington
John Adams
Thomas Jefferson
James Madison
James Monroe
John Quincy Adams
Andrew Jackson
Martin Van Buren
William Henry Harrison
John Tyler
James K. Polk
Zachary Taylor
Millard Fillmore
Franklin Pierce
James Buchanan
Abraham Lincoln
Andrew Johnson
Ulysses S. Grant

```
Rutherford B. Hayes
James A. Garfield
Chester A. Arthur
Grover Cleveland
Benjamin Harrison
Grover Cleveland
William McKinley
Theodore Roosevelt
William Howard Taft
Woodrow Wilson
Warren G. Harding
Calvin Coolidge
Herbert Hoover
Franklin D. Roosevelt
Harry S. Truman
Dwight D. Eisenhower
John F. Kennedy
Lyndon B. Johnson
Richard Nixon
Gerald Ford
Jimmy Carter
Ronald Reagan
George H. W. Bush
Bill Clinton
George W. Bush
Barack Obama
Donald Trump
```

[3]:

```
### using the data in https://en.wikipedia.org/wiki/
 ↪List_of_largest_manufacturing_companies_by_revenue
### find the total revenue for each industry

html = urlopen("https://en.wikipedia.org/wiki/
 ↪List_of_largest_manufacturing_companies_by_revenue")
soup = BeautifulSoup(html)

table = soup.find("table", {"class":"wikitable"})

rows = table.find_all("tr")
rows

## extract the column names from the first row

col = [var.get_text().replace("\n", "") for var in rows[0].find_all('th')]

### create an empty dataframe
```

```
df = pd.DataFrame()

### Extract all other rows in the data

for i in range(1, len(rows)):
    values = [value.text.replace("\n", "").replace("\xa0", "") for value in␣
 ↪rows[i].find_all("td")]
    #print(values)
    df = df.append(pd.Series(values), ignore_index = True)

df.columns = col
df

df.columns[3]
df.rename(columns={"Revenue (by US$ million)": "Revenue"}, inplace=True)

# get rid of the commas
df.Revenue = df['Revenue'].str.replace(",", "")
df.head()

# change the type of Revenue from text to numeric
df.Revenue = pd.to_numeric(df.Revenue)

# find the total of revenue by industry
df.groupby("Industry")["Revenue"].sum()
```

```
[3]: Industry
     Aerospace & Defense                    343610
     Aluminium                               46684
     Automotive                            2112759
     Automotive, Electronics                 44785
     Building Materials, Glass               44701
     Building materials                      46002
     Chemicals                              310053
     Construction equipment                  45462
     Consumer goods                         126765
     Electronics                            868147
     Electronics, various                   211940
     Engineering                            102767
     Engineering, various                  1138802
     Food & Beverages                       344415
     Food, Beverages & Tabacco               45794
     Industrial Machinery                    42638
     Luxury goods                            49221
     Metals                                 118387
     Motor Vehicles & Parts                 119482
     Oil & gas                               80006
```

```
Personal care products                            76450
Pharmaceuticals                                  512868
Renewable energy                                  84134
Shipbuilding                                      44431
Steel                                            279731
Telecommunications equipment                      48005
Telecommunications equipment, Electronics         89311
Textiles                                          98766
Tyres                                             49608
Name: Revenue, dtype: int64
```

[ ]: