Web Scraping Exercise : Wikipedia Coronavirus Data

In [1]:
```
# Import Libraries
import requests
from bs4 import BeautifulSoup
import numpy as np
import pandas as pd
```

In [2]:
```
# Get webpage content
url = "https://en.wikipedia.org/wiki/2020_coronavirus_pandemic_in_Spai
n"
page = requests.get(url)
```

In [3]:
```
# HTML Parser using Beautiful Soup
soup = BeautifulSoup(page.content, 'html.parser')
# print(soup.prettify())
```

Search and Clean Dataframes : Age Tables

In [4]:

```
for i in range(1,4):
    table = soup.find_all('table', class_ = 'wikitable sortable')[i]

    # initialize empty dataframe
    age_table = pd.DataFrame(columns=range(0,10), index = list(range(0
,len(table.find_all('tr'))-3)))

    # Handle column names
    col_names = []
    for row in table.find_all('tr')[:2]:
        cols = row.find_all('th')
        for name in cols:
            col_names.append(name.get_text())

    col_names = [x.replace('\n', '') for x in col_names]

    iterables = [col_names[1:5],col_names[6:8]]


    # iterate through the table label of html
    row_marker = 0
    # row = 0
    for row in table.find_all('tr')[2:len(table.find_all('tr'))-1]:
        column_marker = 0
        groups = row.find('th')
        age_table.iat[row_marker,column_marker] = groups.get_text()
        columns = row.find_all('td')
        for column in columns:
            age_table.iat[row_marker,column_marker+1] = column.get_tex
t()
            column_marker += 1
        row_marker += 1

    columns = pd.MultiIndex.from_product(iterables)
    temp_table = pd.DataFrame(np.array(age_table.iloc[:,1:9]),columns=
columns)
    temp_table.insert(0,col_names[0],np.array(age_table.iloc[:,0]))
    temp_table.insert(9,col_names[5],np.array(age_table.iloc[:,9]))

    if i==1:
        general_age_table = temp_table.replace(r'\n',' ', regex=True)
    elif i == 2:
        women_age_table = temp_table.replace(r'\n',' ', regex=True)
    else:
        men_age_table = temp_table.replace(r'\n',' ', regex=True)
```

In [5]:
```
general_age_table.head()
```

Out[5]:

	Age(years)	Cases		Hospit.		ICU		Deaths		Lethality(%)
		n	%	n	%	n	%	n	%	
0	0-9	433	(0.3)	159	(0.3)	19	(0.4)	1	(0.0)	(0.2)
1	10-19	738	(0.5)	150	(0.2)	8	(0.2)	2	(0.0)	(0.3)
2	20-29	6,864	(5.1)	972	(1.6)	54	(1.0)	20	(0.2)	(0.3)
3	30-39	12,671	(9.3)	2,532	(4.1)	178	(3.5)	37	(0.3)	(0.3)
4	40-49	19,877	(14.6)	5,822	(9.5)	459	(8.9)	118	(1.1)	(0.6)

Search and Clean Dataframes : Pre-existing Factors and Timeline Tables

In [6]:
```
for i in range(4,6):
    table = soup.find_all('table', class_ = ['wikitable sortable','sor
tbottom'])[i]

    table_data = pd.DataFrame(columns=range(0,3), index = list(range(0
,len(table.find_all('tr'))-2)))

    # Handle Data
    row_marker = 0
    for row in table.find_all('tr')[1:len(table.find_all('tr'))-1]:
        column_marker = 0
        columns = row.find_all('td')
        for column in columns:
            table_data.iat[row_marker,column_marker] = column.get_text
()
            column_marker += 1
        row_marker += 1

    # Clean up Null Values
    table_data.dropna(inplace = True)

    # Handle Column Names
    col_names = []
    for row in table.find_all('tr')[:1]:
        cols = row.find_all('th')
        for name in cols:
            col_names.append(name.get_text())

    col_names = [x.replace('\n', '') for x in col_names]

    # Add col names to data
    temp_table = pd.DataFrame(np.array(table_data),columns=col_names)

    if i==4:
        factors_table = temp_table.replace(r'\n',' ', regex=True)
    else:
        timeline_table = temp_table.replace(r'\n',' ', regex=True)
```

In [7]:
```
factors_table.head()
```

Out[7]:

	Diseases and risk factors	% of confirmed	% of deceased
0	Cardiovascular disease	33%	67%
1	Respiratory disease	10%	19%
2	Diabetes	17%	34%
3	Hypertension	14%	N/A

Export data to csv

In [8]:
```
# Create a Pandas Excel writer using XlsxWriter as the engine.
writer = pd.ExcelWriter('coronavirus_spain_data.xls', engine='xlsxwriter')

# Write each dataframe to a different worksheet.
general_age_table.to_excel(writer, sheet_name='Sheet1')
women_age_table.to_excel(writer, sheet_name='Sheet2')
men_age_table.to_excel(writer, sheet_name='Sheet3')
factors_table.to_excel(writer, sheet_name='Sheet4')
timeline_table.to_excel(writer, sheet_name='Sheet5')



# Close the Pandas Excel writer and output the Excel file.
writer.save()
```

References

- https://en.wikipedia.org/wiki/2020_coronavirus_pandemic_in_Spain (https://en.wikipedia.org/wiki/2020_coronavirus_pandemic_in_Spain)
- https://srome.github.io/Parsing-HTML-Tables-in-Python-with-BeautifulSoup-and-pandas/ (https://srome.github.io/Parsing-HTML-Tables-in-Python-with-BeautifulSoup-and-pandas/)
- https://xlsxwriter.readthedocs.io/example_pandas_multiple.html (https://xlsxwriter.readthedocs.io/example_pandas_multiple.html)