Hands-on Lab : Web Scraping

Estimated time needed: **30 to 45** minutes

Objectives

In this lab you will perform the following:

- Extract information from a given web site
- Write the scraped data into a csv file.

Extract information from the given web site

You will extract the data from the below web site:

```
In [4]: https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DA0321EN-SkillsNetwork/labs/datasets/https://
        url = ""
```

```
  File "/tmp/ipykernel_812/1831490351.py", line 1
    https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DA0321EN-SkillsNetwork/labs/datasets/http
s://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DA0321EN-SkillsNetwork/labs/datasets/Programming_La
nguages.htmlProgramming_Languages.html
           ^
SyntaxError: invalid syntax
```

```
In [ ]:
```

The data you need to scrape is the **name of the programming language** and **average annual salary**.
It is a good idea to open the url in your web broswer and study the contents of the web page before you start to scrape.

Import the required libraries

In [ ]:

```
from bs4 import BeautifulSoup
import requests
```

Download the webpage at the url

In [ ]:

```
url = "http://www.ibm.com"
page = requests.get(url)
soup = BeautifulSoup(page.text, 'html')
print(soup)
```

Create a soup object

In [ ]:

```
import requests
from bs4 import BeautifulSoup

# Define the URL
url = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DA0321EN-SkillsNetwork/labs/datasets/Pr

# Send an HTTP GET request to fetch the webpage content
response = requests.get(url)

# Parse the content using BeautifulSoup
soup = BeautifulSoup(response.content, "html.parser")

# Display the parsed content (optional)
print(soup.prettify())
```

Scrape the `Language name` and `annual average salary`.

In [5]:

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

```python
# Define the URL
url = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DA0321EN-SkillsNetwork/labs/datasets/Pr

# Fetch the page content
response = requests.get(url)

# Parse the content using BeautifulSoup
soup = BeautifulSoup(response.content, 'html.parser')

# Extract the table data
table = soup.find('table')

# Initialize lists to store the data
languages = []
salaries = []

# Loop through the table rows
for row in table.find_all('tr')[1:]:
    cols = row.find_all('td')
    languages.append(cols[0].text.strip())
    salaries.append(cols[1].text.strip())

# Create a DataFrame
data = pd.DataFrame({
    'Language': languages,
    'Average Salary': salaries
})

# Display the scraped data
print(data)
```

```
  Language Average Salary
0        1         Python
1        2           Java
2        3              R
3        4     Javascript
4        5          Swift
5        6            C++
6        7             C#
7        8            PHP
8        9            SQL
9       10             Go
```

Save the scrapped data into a file named *popular-languages.csv*

In [6]:

```
import csv

# Example data to be saved
data = [
    ['Language', 'Popularity'],
    ['Python', 'High'],
    ['JavaScript', 'High'],
    ['Java', 'Medium'],
    ['C++', 'Medium']
]

# Specify the file name
filename = 'popular-languages.csv'

# Writing to csv file
with open(filename, mode='w', newline='') as file:
    writer = csv.writer(file)
    writer.writerows(data)

print(f"Data has been saved to {filename}")
```

```
Data has been saved to popular-languages.csv
```

In [7]:

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

# Define the URL
url = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DA0321EN-SkillsNetwork/labs/datasets/Pr

# Fetch the page content
response = requests.get(url)

# Parse the content using BeautifulSoup
soup = BeautifulSoup(response.content, 'html.parser')

# Extract the table data
table = soup.find('table')
```

```
# Initialize lists to store the data
languages = []
salaries = []

# Loop through the table rows
for row in table.find_all('tr')[1:]:
    cols = row.find_all('td')
    languages.append(cols[0].text.strip())
    salaries.append(cols[1].text.strip())

# Create a DataFrame
data = pd.DataFrame({
    'Language': languages,
    'Average Salary': salaries
})

# Display the scraped data
print(data)
```

```
  Language Average Salary
0        1         Python
1        2           Java
2        3              R
3        4     Javascript
4        5          Swift
5        6            C++
6        7             C#
7        8            PHP
8        9            SQL
9       10             Go
```

Authors

Ramesh Sannareddy

Other Contributors

Rav Ahuja

Change Log

Date (YYYY-MM-DD)	Version	Changed By	Change Description
2020-10-17	0.1	Ramesh Sannareddy	Created initial version of the lab