AAM1 Task 1: Web Scraping

RathiPriyanka Srinivasan

Department of Information Technology, Western Governors University

C996: Programming in Python

Dr. Emelda Ntinglet

May 13, 2020

A. The HTML code is extracted using the 'urllib.requests' function (The Python Software Foundation, 2020) from the given URL, https://www.census.gov/programs-surveys/popest.html, (*The Census Bureau's Population Estimates Program*, 2020) and converted to text.

Part A Program Code

```
from bs4 import BeautifulSoup
from urllib.parse import urljoin
import requests
import csv
import re


url = 'https://www.census.gov/programs-surveys/popest.html'
html_url = requests.get(url).text
soup = BeautifulSoup(html_url, "lxml")
```

B. The Python program uses the Beautiful Soup library (Richardson, 2019) to parse through the HTML code extracted from the previous function. By using the **soup.find_all('a')** function the program searches through the HTML code for hyperlinks by searching for the '<a>' tag (W3Schools, 2020) and then only adding locator links to 'url_list' through **url_list.append(l.get('href'))** (Programiz, 2020).

Part B Program Code

```
url_list = []

for l in soup.find_all('a'):
    url_list.append(l.get('href'))
```

C. Each link within 'mylist' is categorized as absolute (starts with "http") or relative (starts with "/"). Absolute links are added to the 'abs_URL' list as is and relative links are added to the 'rel_URL' list while adding the base URL, https://www.census.gov, to create an absolute URL.

Both ‘abs_URL’ and ‘rel_URL’ are concatenated to create the ‘results’ list, **results = abs_URL + rel_URL**.

Part C Program Code

```
for elem in mylist:
if isinstance(elem, str):

    if elem.startswith("http"):
        abs_link = elem
        abs_URL.append(abs_link)
    else:
        rel_link = "https://www.census.gov" + elem
        rel_URL.append(rel_link)
```

D. A new list is created, ‘mylist, and the program checks if each link in ‘url_list’ exists within ‘mylist’. If the link does not already exist within ‘mylist’ then it is added to ‘mylist’ (Geek for Geeks, 2020). To avoid the **CSVWriter** from writing duplicate links, each element in the ‘abs_URL’ list is checked to see if it matches any element from the ‘rel_URL’ list. If the element *does* match, then it is removed from the ‘results’ list. Anchor tags (Ryte, 2019) are removed by searching each element, ‘r’, in the ‘results’ list and then writing the row with **writer.writerow([r])** if ‘r’ does not contain the string “.gov#”.

Part D Program Code

```
for d in url_list:
    if d not in mylist:
        mylist.append(d)
```

Part D Program Code – CSVWriter

```
for j in abs_URL:
    if j in rel_URL:
        dup = j
        results.remove(dup)

for r in results:
    if ".gov#" not in r:
        writer.writerow([r])
```

E. The Python web scraper program code for extracting all *unique* links from the given URL,

https://www.census.gov/programs-surveys/popest.html, is given below.

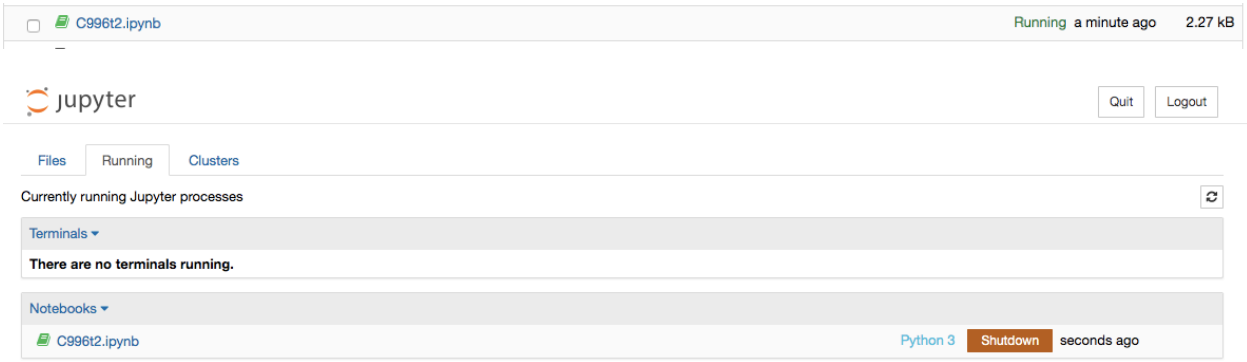
Part E Program Code

```
from bs4 import BeautifulSoup
from urllib.parse import urljoin
import requests
import csv
import re


url = 'https://www.census.gov/programs-surveys/popest.html'
html_url = requests.get(url).text
soup = BeautifulSoup(html_url, "lxml")

url_list = []

for l in soup.find_all('a'):
    url_list.append(l.get('href'))
    mylist = []
    for d in url_list:
        if d not in mylist:
            mylist.append(d)

    with open("C996t1.csv", "w", newline="") as f:
        writer = csv.writer(f, delimiter = ',')
        abs_URL = []
        rel_URL = []
        results = []

        for elem in mylist:
            if isinstance(elem, str):

                if elem.startswith("http"):
                    abs_link = elem
                    abs_URL.append(abs_link)
                else:
                    rel_link = "https://www.census.gov" + elem
                    rel_URL.append(rel_link)

        results = abs_URL + rel_URL

        for j in abs_URL:
            if j in rel_URL:
                dup = j
                results.remove(dup)

        for r in results:
            if ".gov#" not in r:
                writer.writerow([r])
```

F. The HTML code for the provided URL, https://www.census.gov/programs-surveys/popest.html, is located in the 'CensusHTML.htm' file.

G. The output CSV file is located in the 'C996t2.csv' file.

H. A screenshots of the script running to completion is provided below:




```
[I 20:29:56.552 NotebookApp] Starting buffering for e23293de-0800-446f-bd3b-df9ebc9ec983:3107aa374d534870aec63d7b28e8b402
[I 20:31:44.253 NotebookApp] Saving file at /C996t2.ipynb
[I 20:42:59.162 NotebookApp] Saving file at /C996t2.ipynb
```

References

Geeks for Geeks. (2020). loops in python. GeeksforGeeks. https://www.geeksforgeeks.org/loops-in-python/

Programiz. (2020). Python List append(). Parewa Labs Pvt, Ltd. https://www.programiz.com/python-programming/methods/list/append

Richardson, Leonard. (2019). *Beautiful Soup Documentation*. Sphinx. https://www.crummy.com/software/BeautifulSoup/bs4/doc/#getting-help

Ryte Wiki. (2019). *Anchor Tag*. Ryte. https://en.ryte.com/wiki/Anchor_Tag

The Census Bureau's Population Estimates Program. (2020). *Population and Housing Estimates.* United States Census Bureau. https://www.census.gov/programs-surveys/popest.html#

The Python Software Foundation. (2020). *urllib.request — Extensible library for opening URLs*. The Python Standard Library. https://docs.python.org/3/library/urllib.request.html

W3Schools. (2020). HTML <a> Tag. Refsnes Data. https://www.w3schools.com/tags/tag_a.asp