Chapter 5: Scraping Data From Multiple Web pages

DR. LINDA MAHMOUDI

”يرفعِ اللهُ الذين أمنوا منكم والذين أوتوا العلم درجات“

In this chapter, we'll learn how to:

- Scrape **Multiple Web Pages** Using Python.

Scraping Data From Multiple Web pages

- get data from multiple pages from the same website or multiple different URLs as well,
- manually writing code for each webpage is a time-consuming and tedious task.
- Plus, it defines all basic principles of automation. Duh!

Scraping Data From Multiple Web pages

- To solve this exact problem, we will see two main techniques that will help us extract data from multiple webpages:
  - The same website
  - Different website URLs

Scrape Data From the same Website

Now, using the above code, we can get the titles of all the articles by just sandwiching those lines with a loop

```
import requests
from bs4 import BeautifulSoup as bs

URL = 'https://www.geeksforgeeks.org/page/'

for page in range(1,10):
      # pls note that the total number of
    # pages in the website is more than 5000 so i'm only taking the
    # first 10 as this is just an example

    req = requests.get(URL + str(page) + '/')
    soup = bs(req.text, 'html.parser')

    titles = soup.find_all('div',attrs={'class','head'})

    for i in range(4,19):
        if page>1:
            print(f"{(i-3)+page*15}" + titles[i].text)
        else:
            print(f"{i-3}" + titles[i].text)
```

ex, https://www.geeksforgeeks.org/page/3/

```
149
How to Build a WiFi Scanner in Python using Scapy?



150
Pretty-Printing in BeautifulSoup



Process finished with exit code 0
```

Output for the above code

Note: The above code will fetch the first 10 pages from the website and scrape all the 150 titles of the articles that fall under those pages.

Example 2: The link to the website shown above is subslikescript.com/movies.

I also defined a root variable that will help us scrape multiple pages later.

```
root = 'https://subslikescript.com'
website = f'{root}/movies'
```

```
for link in links:
    result = requests.get(f'{root}/{link}')
    content = result.text
    soup = BeautifulSoup(content, 'lxml')
```

As you might remember, the links we stored previously don't contain the root `subslikescript.com`, so we have to concatenate it with the expression `f'{root}/{link}'`.

In case you want to navigate through the pages listed on the web, you have two options:

•**Option 1:** Inspect any of the pages displayed on the website (e.g. 1,2,3, …1234). You should obtain an a tag that contains an href attribute with the links for each page. Once you have the links, concatenate them with the root and follow the steps shown in Section 2.

•**Option 2:** Go to page 2 and copy the link obtained. It should look like this: **subslikescript.com/movies?page=2.** As you can see, there's a pattern the website follows for each page: **f'{website}?page={i}'.** You can reuse the website variable and loop between the numbers 1 and 10 if you want to navigate the first ten pages.

Scrape Data From different Website URLs

Looping through a list of different URLs

```
import requests
from bs4 import BeautifulSoup as bs
URL =
['https://www.geeksforgeeks.org','https://www.geeksforgeeks.org/page/10/']

for url in range(0,2):
    req = requests.get(URL[url])
    soup = bs(req.text, 'html.parser')

    titles = soup.find_all('div',attrs={'class','head'})
    for i in range(4, 19):
        if url+1  > 1:
            print(f"{(i - 3) + url * 15}" + titles[i].text)
        else:
            print(f"{i - 3}" + titles[i].text)
```