

## Web Scraping-2



### What is our GOAL for this MODULE?

The goal of this module is to scrape more data from the exoplanets link.

### What did we ACHIEVE in the class TODAY?

- We scraped the stars data from Nasa's site

### Which CONCEPTS/CODING BLOCKS did we cover today?

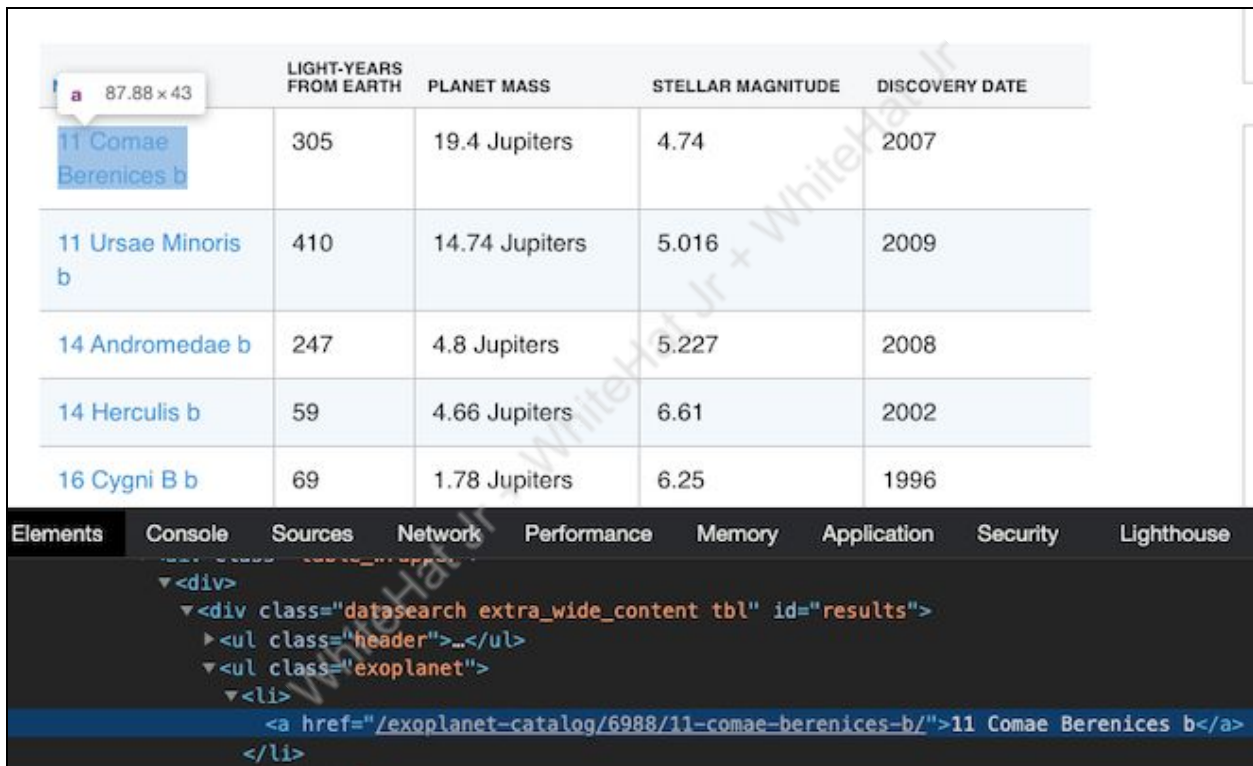
- Usage of selenium
- Usage of BeautifulSoup
- Getting data from HTML of a page

### How did we DO the activities?

1. We created a virtual environment.
2. We installed the necessary libraries.
3. We imported the selenium and BeautifulSoup in our code.
4. We added a new hyperlink to the header.

```
headers = ["name", "light_years_from_earth", "planet_mass", "stellar_magnitude",
"discovery_date", "hyperlink"]
```

5. We checked the hyperlinks.



The screenshot displays a web browser window showing a table of exoplanets. A tooltip indicates the table dimensions as 87.88 x 43. The table has five columns: Name, Light-Years from Earth, Planet Mass, Stellar Magnitude, and Discovery Date. The first row is highlighted, showing '11 Comae Berenices b' with a light-year distance of 305, a mass of 19.4 Jupiters, a stellar magnitude of 4.74, and a discovery date of 2007. Below the table, the browser's developer tools are open to the 'Elements' tab, showing the HTML structure of the table. The first row is highlighted in blue, showing the following HTML code:

```
<div>
  <div class="datasearch extra_wide_content tbl" id="results">
    <ul class="header">...</ul>
    <ul class="exoplanet">
      <li>
        <a href="/exoplanet-catalog/6988/11-comae-berenices-b/">11 Comae Berenices b</a>
      </li>
    </ul>
  </div>
```

6. We saw that the <https://exoplanets.nasa.gov> was missing so we coded to add it.

```

scraper_2.py
1  from selenium import webdriver
2  from bs4 import BeautifulSoup
3  import time
4  import csv
5
6  START_URL = "https://exoplanets.nasa.gov/exoplanet-catalog/"
7  browser = webdriver.Chrome("/Users/apoorvelous/Downloads/chromedriver")
8  browser.get(START_URL)
9  time.sleep(10)
10
11 def scrape():
12     headers = ["name", "light_years_from_earth", "planet_mass", "stellar_magnitude", "discovery_date", "hyperlink"]
13     planet_data = []
14     for i in range(0, 428):
15         soup = BeautifulSoup(browser.page_source, "html.parser")
16         for ul_tag in soup.find_all("ul", attrs={"class", "exoplanet"}):
17             li_tags = ul_tag.find_all("li")
18             temp_list = []
19             for index, li_tag in enumerate(li_tags):
20                 if index == 0:
21                     temp_list.append(li_tag.find_all("a")[0].contents[0])
22                 else:
23                     try:
24                         temp_list.append(li_tag.contents[0])
25                     except:
26                         temp_list.append("")
27             hyperlink_li_tag = li_tags[0]
28             temp_list.append("https://exoplanets.nasa.gov"+hyperlink_li_tag.find_all("a", href=True)[0]["href"])
29             planet_data.append(temp_list)
30             browser.find_element_by_xpath('//*[@id="primary_column"]/footer/div/div/div/nav/span[2]/a').click()
31     with open("scraper_2.csv", "w") as f:
32         csvwriter = csv.writer(f)
33         csvwriter.writerow(headers)
34         csvwriter.writerows(planet_data)
35
36     scrape()
37

```

7. We coded to get the data from the hyperlink.

```

scraper_2.py
1  from selenium import webdriver
2  from bs4 import BeautifulSoup
3  import requests
4  import time
5  import csv
6
7  START_URL = "https://exoplanets.nasa.gov/exoplanet-catalog/"
8  browser = webdriver.Chrome("/Users/apoorvelous/Downloads/chromedriver")
9  browser.get(START_URL)
10 time.sleep(10)
11 headers = ["name", "light_years_from_earth", "planet_mass", "stellar_magnitude", "discovery_date", "hyperlink", "planet_type", "planet_r"]
12 planet_data = []
13 new_planet_data = []
14
15 def scrape():
16     for i in range(0, 428):
17         soup = BeautifulSoup(browser.page_source, "html.parser")
18         for ul_tag in soup.find_all("ul", attrs={"class", "exoplanet"}):
19             li_tags = ul_tag.find_all("li")
20             temp_list = []
21             for index, li_tag in enumerate(li_tags):
22                 if index == 0:
23                     temp_list.append(li_tag.find_all("a")[0].contents[0])
24                 else:
25                     try:
26                         temp_list.append(li_tag.contents[0])
27                     except:
28                         temp_list.append("")
29             hyperlink_li_tag = li_tags[0]
30             temp_list.append("https://exoplanets.nasa.gov"+hyperlink_li_tag.find_all("a", href=True)[0]["href"])
31             planet_data.append(temp_list)
32             browser.find_element_by_xpath('//*[@id="primary_column"]/footer/div/div/div/nav/span[2]/a').click()
33
34 def scrape_more_data(hyperlink):
35     page = requests.get(hyperlink)
36     soup = BeautifulSoup(page.content, "html.parser")
37     for tr_tag in soup.find_all("tr", attrs={"class": "fact_row"}):
38         td_tags = tr_tag.find_all("td")
39         temp_list = []
40         for td_tag in td_tags:
41             try:
42                 temp_list.append(td_tag.find_all("div", attrs={"class": "value"})[0].contents[0])
43             except:
44                 temp_list.append("")
45         new_planet_data.append(temp_list)
46
47 scrape()
48 for data in planet_data:
49     scrape_more_data(data[5])
50
51 final_planet_data = []
52
53 for index, data in enumerate(planet_data):
54     final_planet_data.append(data + final_planet_data[index])
55
56 with open("final.csv", "w") as f:
57     csvwriter = csv.writer(f)
58     csvwriter.writerow(headers)
59     csvwriter.writerows(final_planet_data)

```

**What's NEXT?**

In the next class, we will explore more of scraping and data cleaning.

**EXTEND YOUR KNOWLEDGE:**

You can read the following blog on scraping with selenium to understand more:

<https://medium.com/y medialabs-innovation/web-scraping-using-beautiful-soup-and-selenium-for-dynamic-page-2f8ad15efe25>

WhiteHat Jr + WhiteHat Jr + WhiteHat Jr