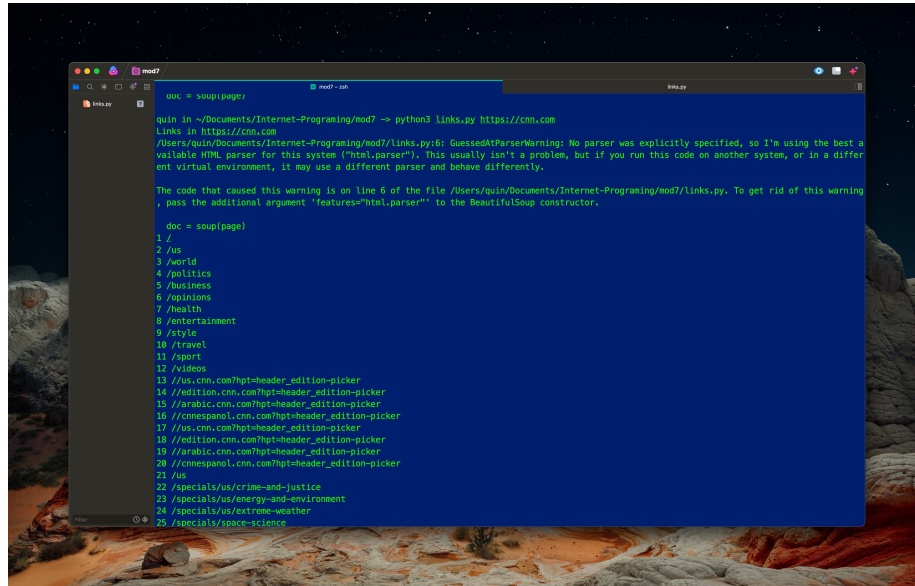


Module 7

Quindarius Lyles-Woods



```
doc = soup(page)
Links in https://cnn.com
/Users/quindarius/Documents/Internet-Programming/mod7/links.py:6: GuessedAtParserWarning: No parser was explicitly specified, so I'm using the best
available HTML parser for this system ('html.parser'). This usually isn't a problem, but if you run this code on another system, or in a differ
ent virtual environment, it may use a different parser and behave differently.

The code that caused this warning is on line 6 of the file /Users/quindarius/Documents/Internet-Programming/mod7/links.py. To get rid of this warning
, pass the additional argument 'features='html.parser'' to the BeautifulSoup constructor.

doc = soup(page)
1 /
2 /us
3 /world
4 /politics
5 /business
6 /opinions
7 /health
8 /entertainment
9 /style
10 /travel
11 /sport
12 /videos
13 //us.cnn.com/hpt-header_edition-picker
14 //edition.cnn.com/hpt-header_edition-picker
15 //arabic.cnn.com/hpt-header_edition-picker
16 //cnnespanol.cnn.com/hpt-header_edition-picker
17 //us.cnn.com/hpt-header_edition-picker
18 //edition.cnn.com/hpt-header_edition-picker
19 //arabic.cnn.com/hpt-header_edition-picker
20 //cnnespanol.cnn.com/hpt-header_edition-picker
21 /us
22 /specials/us/crime-and-justice
23 /specials/us/energy-and-environment
24 /specials/us/extreme-weather
25 /specials/space-science
```

Figure 1: Output from Terminal

Report

Installing the package request and beautiful soup allowed me to run a web scraper on any site that I use the command-line to output.

Code for the scraper

```
def get_links(url):
    import requests
    from bs4 import BeautifulSoup as soup
    result = requests.get(url)
    page = result.text
    doc = soup(page)
    links = [element.get('href') for element in doc.find_all('a')]
    return links

if __name__ == '__main__':
    import sys
    for url in sys.argv[1:]:
        print('Links in', url)
```

```
for num, link in enumerate(get_links(url), start=1):  
    print(num, link)  
print()
```