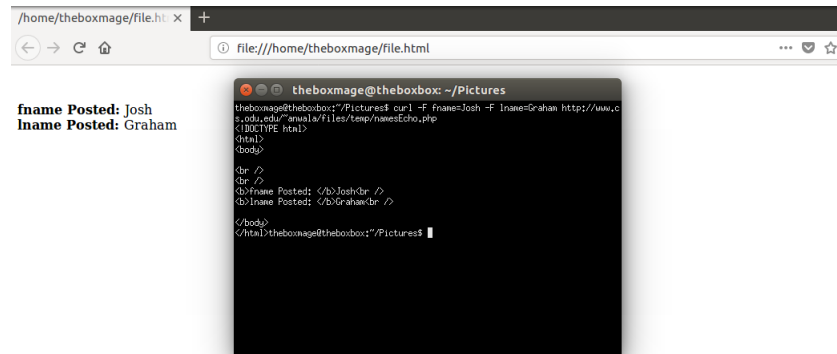


1. The below screenshot shows both the result pushed to a webpage, as well as the output from running the command on the commandline.



2. Examples of how its run can be found in the txt files web1.txt web2.txt and web3.txt in the same directory as this file.

```
import sys
import re
import requests
import urllib.request
from bs4 import BeautifulSoup

#Checking for at least one command line argument
if len(sys.argv) == 1:
    print('Not enough arguments')
    sys.exit()

#Downloading the information from the URL
resp = urllib.request.urlopen(sys.argv[1])
#Creating the BeautifulSoup Object to parse the webpage
soup = BeautifulSoup(resp, "html.parser")
print("Given URL: " )
print(sys.argv[1])
#Looping through all the links
for link in soup.find_all('a', attrs={'href': re.compile("^http:")}):
    #This cleanly and elegantly deals with any redirects.
    r = requests.head(link['href'], allow_redirects=True)
    # If the content type is application/pdf, its a pdf.
    if r.headers["Content-Type"] == "application/pdf":
        print("URL: ")
        print(r.url)
        print("Length in bytes: ")
        print(r.headers["Content-Length"])
        print()
```

3. The disconnected section really doesn't need an explanation. Tubes and tendrils only had one entry each, and they follow the path from the left value to the right one. The out values were harder for me, but they both stand relatively free. My largest issue was trying to figure out the difference between In and Scc. I'm still not overly confident that they are correct, but this is the closest I could get given the analogy.

IN : I, P, O, M, N, L

SCC : A, B, C, G

OUT : H, D

TENDRILS : I, K

TUBES : L, D

DISCONNECTED : E, F