

**Bo Zhang**  
**01063214**

**Task 1:**

Demonstrate that you know how to use "curl" well enough to correctly POST data to a form. Show that the HTML response that is returned is "correct". That is, the server should take the arguments you POSTed and build a response accordingly. Save the HTML response to a file and then view that file in a browser and take a screen shot.

**Curl command:** `curl -d "firstname=Roy&lastname=Zhang" http://quiet-waters-1228.herokuapp.com/echo`

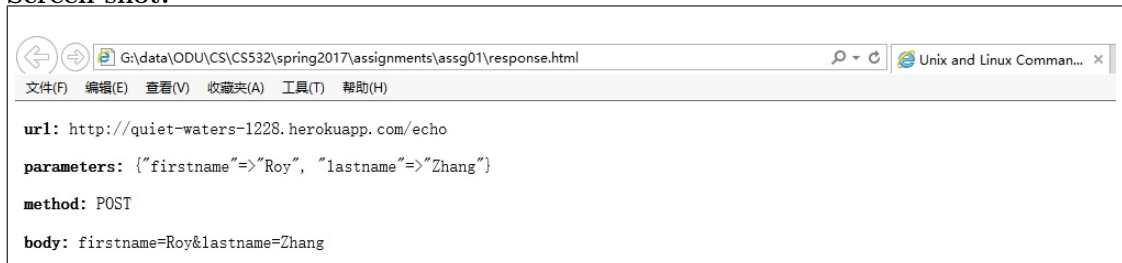
**HTML response:**

```
<!DOCTYPE html>
<html>
<head>
  <title>Unix and Linux Commands for Developers</title>
  <link href="/assets/application.css" media="all" rel="stylesheet" />
  <script src="/assets/application.js"></script>
</head>
<body>

  <p>
    <strong>url:</strong>
    <span>
      http://quiet-waters-1228.herokuapp.com/echo
    </span>
  </p>
  <p>
    <strong>parameters:</strong>
    <span>
      {"firstname"=>"Roy", "lastname"=>"Zhang"}
    </span>
  </p>
  <p>
    <strong>method:</strong>
    <span>
      POST
    </span>
  </p>
  <p>
    <strong>body:</strong>
    <span>
      firstname=Roy&lastname=Zhang
    </span>
  </p>
```

</p>  
</body>  
</html>

#### Screen shot:



**Note:** I found this website from "http://conqueringthecommandline.com/book/curl"

**Task 2:** Write a Python program that:

1. takes as a command line argument a web page
2. extracts all the links from the page
3. lists all the links that result in PDF files, and prints out the bytes for each of the links. (note: be sure to follow all the redirects until the link terminates with a "200 OK".)
4. show that the program works on 3 different URIs, one of which needs to be:  
`http://www.cs.odu.edu/~mln/teaching/cs532-s17/test/pdfs.html`

#### Algorithm:

1. Ask to input a URI
2. Open this URI and save it into an html object
3. Extracts all the links from the html object
4. Open all the links 1 by 1 and check their Content-Type from the response
5. If the Content-Type is PDF, get the URI from the response of the opened link and print it
6. Print the Content-Length from the response of the opened link

#### Results:

1. `http://www.cs.odu.edu/~mln/teaching/cs532-s17/test/pdfs.html`

```
Python 3.5.2 Shell
File Edit Shell Debug Options Window Help
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 25 2016, 22:01:18) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
== RESTART: G:\data\ODU\CS\CS532\spring2017\assignments\assg01\al_bzhang.py ==
Please enter a URI
http://www.cs.odu.edu/~mln/teaching/cs532-s17/test/pdfs.html
http://www.cs.odu.edu/~mln/pubs/ht-2015/hypertext-2015-temporal-violations.pdf
2184076
http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-annotations.pdf
622981
https://arxiv.org/pdf/1512.06195.pdf
1748961
http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-off-topic.pdf
4308768
http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-stories.pdf
1274604
http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-profiling.pdf
639001
http://www.cs.odu.edu/~mln/pubs/jcdl-2014/jcdl-2014-brunelle-damage.pdf
2205546
http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-temporal-intention.pdf
720476
http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-mink.pdf
1254605
http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-arabic-sites.pdf
709420
http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-dictionary.pdf
2350603
>>> |
```

2. <http://www.cs.odu.edu/~mln/>

```
Python 3.5.2 Shell
File Edit Shell Debug Options Window Help
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 25 2016, 22:01:18) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
== RESTART: G:\data\ODU\CS\CS532\spring2017\assignments\assg01\al_bzhang.py ==
Please enter a URI
http://www.cs.odu.edu/~mln/
http://www.cs.odu.edu/~mln/cv.pdf
363963
http://www.cs.odu.edu/~mln/nsf-cv-2014.pdf
88700
http://www.cs.odu.edu/~mln/mln-ad.pdf
92868
>>> |
```

3. <http://www.cs.odu.edu/~mln/teaching/>

```
Python 3.5.2 Shell
File Edit Shell Debug Options Window Help
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 25 2016, 22:01:18) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
== RESTART: G:\data\ODU\CS\CS532\spring2017\assignments\assg01\al_bzhang.py ==
Please enter a URI
http://www.cs.odu.edu/~mln/teaching/
http://www.cs.odu.edu/~mln/pubs/phd/alnoamany-phd-dissertation.pdf
69827914
http://www.cs.odu.edu/~mln/pubs/phd/brunelle-phd-dissertation.pdf
41389855
http://www.cs.odu.edu/~mln/pubs/phd/salaheldeen-phd-dissertation.pdf
10875917
http://www.cs.odu.edu/~mln/pubs/phd/alsum-phd-dissertation.pdf
23475427
http://www.cs.odu.edu/~mln/pubs/phd/cartledge-phd-dissertation.pdf
20240922
http://www.cs.odu.edu/~mln/pubs/phd/klein-phd-dissertation.pdf
7873533
http://www.cs.odu.edu/~mln/pubs/phd/smith-phd-dissertation.pdf
4018713
http://www.harding.edu/fmccown/pubs/lazy-preservation-dissertation.pdf
4301157
http://www.cs.odu.edu/~mln/pubs/ms/jones-ms-2015.pdf
21131710
http://www.cs.odu.edu/~mln/pubs/ms/alam-ms-2013.pdf
1849955
http://www.cs.odu.edu/~mln/pubs/ms/haq-ms-2008.pdf
622509
http://www.cs.odu.edu/~mln/pubs/ms/harrison-ms-2005.pdf
2094637
>>> |
```

**Task 3:** Consider the "bow-tie" graph in the Broder et al. paper (fig 9):  
<http://www9.org/w9cdrom/160/160.html>

Now consider the following graph:

```
A -> B
B -> C
C -> D
C -> A
C -> G
E -> F
G -> C
G -> H
I -> H
I -> K
L -> D
M -> A
M -> N
N -> D
O -> A
```

$P \rightarrow G$

For the above graph, give the values for:

IN: 3 (M, O, P)

SCC: 4 (A, B, C, G)

OUT: 2 (D, H)

Tendrils: 4 (N, L, I, K)

Tubes: 1 (M  $\rightarrow$  N  $\rightarrow$  D)

Disconnected: 2 (E, F)