

STA312H5S Tutorial 1

Matthew C. Scicluna

University of Toronto Mississauga

January 19, 2015

What We Are Going to Cover

A bit more information about this

Today we are going to go over assignment 1 and talk about the first project

Assignment 1 Answers

Goal was to make a web scraper using Python with BeautifulSoup4 package.

Assignment 1 Answers



deathrowinfo.py - C:\Users\Matthew Scicluna\Desktop\Files\TA work\STA313 TA\Assignment1\deathrowinfo.py

File Edit Format Run Options Windows Help

```
import urllib.request
from bs4 import BeautifulSoup
import csv
import time

# get the webpage with the prisoner data
uri='http://www.tdcj.state.tx.us/death_row/dr_executed_offenders.html'

f = csv.writer(open("Death Row Data" + ".csv", "w", newline=''), dialect='excel')

# Write column headers as the first line
f.writerow(["Last Statement", "Last Name", "First Name", "TDCJ Num", "Age", "Date", "Race", "County"])

urllines = urllib.request.urlopen(uri)
pagedat = urllines.read()
urllines.close()
soup = BeautifulSoup(pagedat)
allrows = soup.find_all("tr")
suburi = uri[:38]
for row in allrows:
    tds = row.find_all("td")
    try:
        links=row.find_all('a')
        link=links[1].get('href')
        LSLink = suburi+link
        urllines2 = urllib.request.urlopen(LSLink)
        pagedat2 = urllines2.read()
        urllines2.close()
        soup2 = BeautifulSoup(pagedat2)
        par = soup2.find_all("p")
        for i in range(1, (len(par)-1)):
            if 'Last Statement:' in par[i].get_text():
                LS=str(par[i+1].get_text())
                LS=LS.replace('\x92s','')
                LS=LS.replace('\xa0','')

        LN = str(tds[3].get_text())
        FN = str(tds[4].get_text())
        TDCJ = str(tds[5].get_text())
        Age = str(tds[6].get_text())
```

Assignment 1 Answers

Firstly needed to find all anchor tags in the HTML. For each row from the table I used

```
row.find_all('a') to extract this.
```

This gives me an object links which has this in it

```
>>>links
[<a href="dr_info/brookscharlie.html" title=
"Offender Information">Offender Information</a>,
<a href="dr_info/brookscharlielast.html"
title="Last Statement">Last Statement</a>]
```

Each link object had many links in it, so I used We notice that this object is a list so we just index it to get the desired link...

```
>>> links[1]
<a href="dr_info/brookscharlielast.html"
title="Last Statement">Last Statement</a>
```

We use the following command to pull out the href from the stuff we dont want.

```
links[1].get('href')
```

Now I concatenated this with the website URL to build the address to where the last statement was. I called this LSLink. I then ran

```
urllib.request.urlopen(LSLink)
```

Now I just ran BeautifulSoup on this page in an analogous way as I did on lines 14 to 17, except for 2 minor differences.

1. the page had no table so I looked for the paragraph HTML tag, which is where the last statement was

```
par = soup2.find_all("p")
```

2. Those pesky unicode characters need to be dealt with

```
for i in range(1,(len(par)-1)):
    if 'Last Statement:' in par[i].get_text():
        LS=str(par[i+1].get_text())
        LS=LS.replace('\x92s','')
        LS=LS.replace('\xa0','')
#I stored the last statements in LS
```

Why Talk About Assignment 1 So Much?

I modified the solution to project 1 when making this assignment.
So if you understand the assignment you will understand the project

Here is my hint for project 1. Look at the original web address of the page

```
http://www.diamondse.info/
```

Notice how it changes when we select the Princess diamond

```
http://www.diamondse.info/diamond-prices  
.asp?shape=Princess
```

But when we change the minimum Carat it doesn't seem to affect the webpage's URL...

```
http://www.diamondse.info/diamond-prices.  
asp?shape=Princess&minCarat=0
```