Vaibhav Verma (/)

College Student and Hacker

Scrape the web using CSS Selectors in Python

05 January 2014

Web Scraping (http://en.wikipedia.org/wiki/Web_scraping) is a super useful technique that lets you get data out of web pages that don't have an API. I often scrape web pages to get structured data out of unstructured web pages, and Python is my language of choice for quick scripts.

BeautifulSoup - Why I don't use it anymore

In the past, I used Beautiful Soup (http://www.crummy.com/software/BeautifulSoup/) almost exclusively to do this kind of scraping. BeautifulSoup is a great library for web scraping - it has great docs, and it gets the job done most of the time. I've used it on lots of projects. However, I find that it doesn't fit my workflow.

Let's say I wanted to scrape some data off a web page. I usually inspect the element in the Chrome Dev Console, and guess at a selector that might give me the data I want. Perhaps I guess `div.foo li a`. I quickly check to see if this works by running this selector in the console `$('div.foo li a')`, and modify it if it doesn't.

Even after using BeautifulSoup for a while, I find that I have to go back and read the docs to write code that scrapes this selector. I always forget how to select classes in BeautifulSoup's `find_all` method. I don't remember how to write a CSS attribute selector such as `a[href=*foo*]`. It doesn't let me write code at the speed of thought.

lxml.cssselect

LXML (http://lxml.de/) is a robust library for parsing XML and HTML in Python that even BeautifulSoup is built on top of. I don't know much about `lxml`, except that I can use CSS Selectors with it very easily, thanks to lxml.cssselect (http://lxml.de/cssselect.html). Look at the example code below to see how easy this is.

```
import lxml.html
from lxml.cssselect import CSSSelector

# get some html
import requests

r = requests.get('http://url.to.website/')

# build the DOM Tree
tree = lxml.html.fromstring(r.text)

# print the parsed DOM Tree
print lxml.html.tostring(tree)

# construct a CSS Selector
sel = CSSSelector('div.foo li a')

# Apply the selector to the DOM tree.
results = sel(tree)
print results

# print the HTML for the first result.
match = results[0]
print lxml.html.tostring(match)

# get the href attribute of the first result
print match.get('href')

# print the text of the first result.
print match.text

# get the text out of all the results
data = [result.text for result in results]
```

As you can see, it's really easy to use CSS Selectors with Python and lxml. Instead of spending time reading BeautifulSoup docs, spend time writing your application.

Installation of lxml and lxml.cssselect

LXML and CSSSelect are both Python packages that you can install easily via `pip`. In order to install `lxml` via pip you will need `libxml2` and `libxslt`. On a standard Ubuntu installation, you can simply do

```
sudo apt-get install libxml2-dev libxslt1-dev
pip install lxml cssselect
```

Check out the lxml installation page (http://lxml.de/installation.html) and lxml.cssselect (http://lxml.de/cssselect.html) for more information.