Scraping with Scrapy and Selenium

Asked 11 years, 6 months ago Modified 11 years, 6 months ago Viewed 6k times

6

I have a scrapy spider which crawls a site that reloads content via javascript on the page. In order to move to the next page to scrape, I have been using Selenium to click on the month link at the top of the site.

The problem is that, even though my code moves through each link as expected, the spider just scrapes the first month (Sept) data for the number of months and returns this duplicate data.

How can I get around this?

```
from selenium import webdriver

class GigsInScotlandMain(InitSpider):
        name = 'gigsinscotlandmain'
        allowed_domains = ["gigsinscotland.com"]
        start_urls = ["http://www.gigsinscotland.com"]


    def __init__(self):
        InitSpider.__init__(self)
        self.br = webdriver.Firefox()

    def parse(self, response):
        hxs = HtmlXPathSelector(response)
        self.br.get(response.url)
        time.sleep(2.5)
        # Get the string for each month on the page.
        months = hxs.select("//ul[@id='gigsMonths']/li/a/text()").extract()

        for month in months:
            link = self.br.find_element_by_link_text(month)
            link.click()
            time.sleep(5)

            # Get all the divs containing info to be scraped.
            listitems = hxs.select("//div[@class='listItem']")
            for listitem in listitems:
                item = GigsInScotlandMainItem()
                item['artist'] = listitem.select("div[contains(@class,
'artistBlock')]/div[@class='artistdiv']/span[@class='artistname']/a/text()").extra
                #
                # Get other data ...
                #
                yield item
```

python selenium scrapy

Share Improve this question Follow

asked Sep 16, 2013 at 19:58

puffin
1,351 4 15 18

1 Answer

Sorted by: Highest score (default)

6

The problem is that you are reusing `HtmlXPathSelector` that was defined for the initial response. Redefine it from selenium browser `source_code`:

```
...
for month in months:
    link = self.br.find_element_by_link_text(month)
    link.click()
    time.sleep(5)

    hxs = HtmlXPathSelector(self.br.page_source)

    # Get all the divs containing info to be scraped.
    listitems = hxs.select("//div[@class='listItem']")
...
```

Share Improve this answer Follow

answered Sep 16, 2013 at 20:02

alecxe
474k 127 1.1k 1.2k

Start asking to get answers

Find the answer to your question by asking.

Explore related questions

python selenium scrapy

Ask question

See similar questions with these tags.