Stack Overflow is a community of 4.7 million programmers, just like you, helping each other.

Join them; it only takes a minute:

Join the Stack Overflow community to:

—

Sign up

| Ask programming questions | Answer and help your peers | Get recognized for your expertise |

Scrapy crawl with next page

I have this code for scrapy framework:

```python
# -*- coding: utf-8 -*-
import scrapy
from scrapy.contrib.spiders import Rule
from scrapy.linkextractors import LinkExtractor
from lxml import html

class Scrapy1Spider(scrapy.Spider):
    name = "scrapy1"
    allowed_domains = ["sfbay.craigslist.org"]
    start_urls = (
        'http://sfbay.craigslist.org/search/npo',
    )

    rules = (Rule(LinkExtractor(allow=(), restrict_xpaths=('//a[@class="button next"]',)), callback="parse", follow= True),)

    def parse(self, response):
        site = html.fromstring(response.body_as_unicode())
        titles = site.xpath('//div[@class="content"]/p[@class="row"]')
        print len(titles), 'AAAA'
```

But problem is that i get 100 results, it doesn't go to next pages.

What is wrong here?

python   scrapy   lxml   scrapy-spider

edited Sep 17 '15 at 6:50   asked Sep 17 '15 at 6:43
                                                    Mirza Delic
                                                    621 1 13 39

1 Answer

Your `rule` is not used because you don't use a `CrawlSpider`.

So you have to create the next page `requests` manually like so:

```python
# -*- coding: utf-8 -*-
import scrapy
from scrapy.contrib.spiders import Rule
from scrapy.linkextractors import LinkExtractor
from lxml import html

class Scrapy1Spider(scrapy.Spider):
    name = "craiglist"
    allowed_domains = ["sfbay.craigslist.org"]
    start_urls = (
        'http://sfbay.craigslist.org/search/npo',
    )

    Rules = (Rule(LinkExtractor(allow=(), restrict_xpaths=('//a[@class="button next"]',)), callback="parse", follow= True),)

    def parse(self, response):
        site = html.fromstring(response.body_as_unicode())
        titles = site.xpath('//div[@class="content"]/p[@class="row"]')
        print len(titles), 'AAAA'

        # follow next page links
        next_page = response.xpath('.//a[@class="button next"]/@href').extract()
        if next_page:
            next_href = next_page[0]
            next_page_url = 'http://sfbay.craigslist.org' + next_href
            request = scrapy.Request(url=next_page_url)
            yield request
```

Or use the `CrawlSpider` like so:

```python
# -*- coding: utf-8 -*-
import scrapy
from scrapy.spiders import CrawlSpider, Rule
from scrapy.linkextractors import LinkExtractor
from lxml import html
```

```
class Scrapy1Spider(CrawlSpider):
    name = "craiglist"
    allowed_domains = ["sfbay.craigslist.org"]
    start_urls = (
        'http://sfbay.craigslist.org/search/npo',
    )

    rules = (Rule(LinkExtractor(allow=(), restrict_xpaths=('//a[@class="button
next"]',)), callback="parse_page", follow= True),)

    def parse_page(self, response):
        site = html.fromstring(response.body_as_unicode())
        titles = site.xpath('//div[@class="content"]/p[@class="row"]')
        print len(titles), 'AAAA'
```

edited Sep 17 '15 at 10:11

answered Sep 17 '15 at 9:17

Frank Martin
1,287　2　11　21

can you give me example with `CrawlSpider` ? And is it recommended way? – Mirza Delic Sep 17 '15 at 9:24

Added `CrawlSpider` code. Note that you must not use `parse` as callback function because this is used internally! – Frank Martin Sep 17 '15 at 10:14

Thanks, it works perfect. – Mirza Delic Sep 17 '15 at 10:17