## 20200821信息安全实训

| | | | |
|---|---|---|---|
| **笔记本:** | 我的第一个笔记本 | | |
| **创建时间:** | 2020/8/21 18:19 | **更新时间:** | 2020/8/21 21:05 |
| **作者:** | 820410740@qq.com | | |

全站爬虫 scrapy 爬虫框架
QuotesSpider类

```python
import scrapy

class QuotesSpider(scrapy.Spider):

    name = "quote"

    def start_requests(self):
        urls = [
            "http://quotes.toscrape.com/page/1/",
            "http://quotes.toscrape.com/page/2/",
        ]
        for url in urls:
            yield scrapy.Request(url=url, callback=self.parse)

    def parse(self, response):
        page = response.url.split("/")[-2]
        filename = 'quotes-%s.html' % page
        with open(filename, 'wb') as f:
            f.write(response.body)
        self.log('Saved file %s' % filename)
```

scrapy shell
css
XPath

```python
import scrapy

class LinkSpider(scrapy.Spider):

    name = "links"

    start_urls = [
        'https://www.szlanyou.com',
    ]

    def parse(self, response):
        for a in response.css('a'):
            print(a.css('::attr(href)')href.get())
```

```python
import scrapy

class LinkSpider(scrapy.Spider):

    name = "links"

    start_urls = [
        'https://www.szlanyou.com',
    ]

    def parse(self, response):
        for href in response.css('a::attr(href)').getall():
            print(href)
```

```python
import scrapy

class LinkSpider(scrapy.Spider):

    name = "links"

    start_urls = [
        'https://www.szlanyou.com',
    ]

    def parse(self, response):
        for href in response.css('a'):
            yield {'link': href.css('::attr(href)').get(), }
```

```python
import scrapy
from urllib.parse import urljoin

class LinkSpider(scrapy.Spider):

    name = "links"

    start_urls = [
        'https://www.szlanyou.com',
    ]

    def parse(self, response):
        for href in response.css('a'):
            link = href.css('::attr(href)').get()
            if "https://www.szlanyou.com" not link and link[0] == '/':
                link = urljoin('https://www.szlanyou.com', link)
                yield {'link': link, }
                yield response.follow(link, callback=self.parse)
```

指纹识别

消息摘要算法
MD5 SHA-1

```python
import json
import hashlib
import scrapy
from scrapy.crawler import CrawlerProcess

with open("./links.json", "r") as f:
    raw = f.read()

raw = json.loads(raw)

links = []
for i in raw:
    links.append(i["link"])

fgprts = []
class MySpider(scrapy.Spider):

    name = "fgrprt"

    start_urls = links

    def parse(self, response):
        md5 = hashlib.md5()
        md5.update(response.body)
        finger_print = md5.hexdigest()
        fgprts.append(
            {"url": response.url, "finger_print": finger_print}
        )

process = CrawlerProcess()
process.crawl(MySpider)
process.start()
```

```
osptech@OspMo1:~/tutorial$
```

```python
import json

with open("./fgprts.json", "w") as f:
    f.write(json.dumps(fgprts))
```

管理 控制 视图 热键 设备 帮助

```python
import json
import hashlib
import scrapy
from scrapy.crawler import CrawlerProcess

with open('./fgprts.json') as f:
    fgprts = json.loads(f.read())

urls = [i['url'] for i in fgprts]

results = []
class MySpider(scrapy.Spider):

    name = 'check'

    start_urls = urls

    def parse(self, response):
        for i in fgprts:
            if i['url'] == response.url:
                finger_print = i['finger_print']

        _
```

```
from scrapy.crawler import CrawlerProcess

with open('./fgprts.json', 'r') as f:
    fgprts = json.loads(f.read())

urls = [i['url'] for i in fgprts]

results = []
class MySpider(scrapy.Spider):

    name = 'check'

    start_urls = urls

    def parse(self, response):
        for i in fgprts:
            if i['url'] == response.url:
                finger_print = i['finger_print']
        md5 = hashlib.md5()
        md5.update(response.body)
        finger_print_now = md5.hexdigest()
        results.append(
            {
                'url': response.url,
                'isDistorted': finger_print == finger_print_now
            }
        )

process = CrawlerProcess()
process = crawl(MySpider)
process.start()

with open('./results', 'w') as f:
    f.write('%s %s\n'%())
```

```
process.start()

with open('./results', 'w') as f:
    for i in results:
        f.write('%s %s\n'%(i['url'], str(i['isDistorted'])))
```

```
        )

process = CrawlerProcess()
process.crawl(MySpider)
process.start()
```