crawler Documentation

Egor

Sep 02, 2018

Contents:

User Guide

CHAPTER 1

Installation

At the command line:

```
easy_install crawler
```

Or, if you have `pip` installed:

```
pip install crawler
```

CHAPTER 2

Support

The easiest way to get help with the project is to join the `#crawler` channel on Freenode. We hang out there and you can get real-time help with your projects. The other good way is to open an issue on Github.

The mailing list at is also available for support.

Github

CHAPTER 3

Cookbook

Crawl a web page

The most simple way to use our program is with no arguments. Simply run:

```
python main.py -u <url>
```

to crawl a webpage.

Crawl a page slowly

To add a delay to your crawler, use *-d*:

```
python main.py -d 10 -u <url>
```

This will wait 10 seconds between page fetches.

Crawl only your blog

You will want to use the *-i* flag, which while ignore URLs matching the passed regex:

```
python main.py -i "^blog" -u <url>
```

This will only crawl pages that contain your blog URL.

Programmer Reference

CHAPTER 4

Command Line Options

These flags allow you to change the behavior of Crawler. Check out how to use them in the Cookbook.

-d <sec>, **--delay** <sec>

Use a delay in between page fetchs so we don't overwhelm the remote server. Value in seconds.

Default: 1 second

-i <regex>, **--ignore** <regex>

Ignore pages that match a specific pattern.

Default: None

CHAPTER 5

Crawler Python API

Getting started with Crawler is easy. The main class you need to care about is

crawler.utils.**should_ignore**(*ignore_list*, *url*)
Returns True if the URL should be ignored

Parameters

- **ignore_list** – The list of regexs to ignore.
- **url** – The fully qualified URL to compare against.

```
>>> should_ignore(['blog/$'], 'http://ericholscher.com/blog/')
True

>>> should_ignore(['home'], 'http://ericholscher.com/blog/')
False

>>> log('http://ericholscher.com/blog/', 200)
OK: 200 http://ericholscher.com/blog/

>>> log('http://ericholscher.com/blog/', 500)
ERR: 500 http://ericholscher.com/blog/
```

Other directive is `testcode`

```
log('http://ericholscher.com/blog/', 500)
```

That requires separate `testoutput`

```
ERR: 500 http://ericholscher.com/blog/
```

If i add this text and push will it automatically appear in the docs?

Python Module Index

C

Index

Symbols

C

S