wsinfo Documentation

Release 1.3.0

Linus Groh

September 15, 2016

Contents

Version 1.3.0

Author Linus Groh

Contact mail@linusgroh.de

CHAPTER 1

Contents

1.1 Introduction

1.1.1 In short...

The wsinfo library bundles the power of the `socket` module, some `urllib` subpackages, XML parsing and regular expressions into one library with the possibility to get a huge amount of information for a specific website.

1.1.2 Why should I use it?

Did you ever had to retrieve information about some website? Maybe.

But then you know what a pain it is, if you want to do more than getting the HTML code of a website. You will have to use a lot of different standard and not standard library modules:

Python version	Libraries
Python 2	`urlparse`, `urllib`, `urllib2` and `httplib`
Python 3	`urllib3`, some subpackages of `urllib` and `http`
Both	`socket`, `requests` and `beautifulsoup`

Confused?

While some of the standard library modules were moved or replaced in Python 3 (see above), you will probably have to adapt your code to work under both Python 2 and Python 3.

I don't want to talk about connection issues and the ton of HTTP error codes you'll need to handle one day.

The next step then is parsing the HTML using an HTML or XML parser library, or some difficult regular expressions. Not funny, because some web developers don't care about HTML standards even today.

And that's why you can use the wsinfo library for getting website information on the fly. It really makes your life easier, and your code shorter.

1.1.3 How can I use it?

The library works for both online and localhost websites, it's usage is as easy as:

```
>>> import wsinfo
>>> w = wsinfo.Info("https://github.com")
>>> w.ip
'192.30.253.112'
>>> w.http_status_code
200
>>> w.title
'How people build software · GitHub'
>>> w.content
'<!DOCTYPE html>\n<html>\n[...]\n</html>'
```

Pretty nice, huh?

1.2 Installation

The wsinfo library is available on PyPI, so you can install it using `pip`:

```
pip install wsinfo
```

As an alternative you can get the source code from GitHub and install it using the setup script:

```
python setup.py install
```

Just check the installation:

```
>>> import wsinfo
>>> wsinfo.__version__
'1.3.0'
```

And here we go!

Note: The wsinfo library should be compatible with both Python 2 and 3.

1.3 Usage

1. Make sure you've *installed* the wsinfo library correctly.
2. Run Python and import the library:

```
>>> import wsinfo
```

3. Create an instance of the `Info` class. I'll use the *GitHub* start page in the following examples:

```
>>> w = wsinfo.Info("https://github.com")
```

4. Now you can get all the information:

```
>>> import wsinfo
>>> w = wsinfo.Info("https://github.com")
>>> w.ip
'192.30.253.112'
```

```
>>> w.http_status_code
200
>>> w.title
'How people build software · GitHub'
>>> w.content
'<!DOCTYPE html>\n<html>\n[...]\n</html>'
```

Also see the *API overview* for reference.

Note: All public methods of the `Info` class are using the `@property` decorator, so you'll not have to make function calls. Instead, they're treated as class attributes.

---

5. Full code:

```
import wsinfo

w = wsinfo.Info("https://github.com")
print(w.http_status_code)
print(w.title)
print(w.content)
```

1.4 API

The wsinfo library bundles the power of the `socket` module, some `urllib` subpackages, XML parsing and regular expressions into one library with the possibility to get a huge amount of information for a specific website.

class `wsinfo.`**`Info`**(*url*)

Class collecting some information about the website located at the given URL.

Parameters **`url`** – Valid URL to the website *(e.g. http://example.com/path/to/file.html)*.

`content`

Get the website's content.

Returns Content of the website *(e.g. HTML code)*.

Return type str

`content_type`

Get the website's content type.

Returns Content-type of the website's code *(e.g. text/html)*.

Return type str or NoneType

`favicon_path`

Get the path to the website's icon.

The `href` attribute of the first `<link>` tag containing `rel="icon"` or `rel="shortcut icon"` is used.

Returns The path to the icon of the website *(known as favicon)*.

Return type str or NoneType

`hierarchy`

Get a list representing the heading hierarchy.

Returns List of tuples containing the heading type *(h1, h2, ...)* and the headings text.

Return type list

http_header
Get the website's HTTP header.

Returns HTTP header of the website.

Return type str

http_header_dict
Get the website's HTTP header as dictionary.

Returns HTTP header of the website as dictionary.

Return type dict

http_status_code
Get the website's HTTP status code.

- **1xx:** Information
- **2xx:** Success
- **3xx:** Redirection
- **4xx:** Client error
- **5xx:** Server error

See this Wikipedia article for reference.

Returns HTTP status code of the website.

Return type int

ip
Get the IP address of the website's domain.

Note: This will not always return the IP address of the URL you've passed to the `Info` constructor. For example, the server may redirect to another page, and this function will return the IP address of the redirected URL. If the website implements a client side redirect, you will not be redirected but get the IP address of the URL you've passed before.

Returns IP address of the website's domain.

Return type str

server
Get the server's name/type and version.

Most common are *Apache*, *nginx*, *Microsoft IIS* and *gws* on Google servers.

Returns A list containing the name or type of the server software and (if available) the version number.

Return type list or NoneType

server_country
Get the country the where the server is located.

Warning: This is currently not implemented, I need to do some more research how to do this. I think *whois* is a buzzword...

Returns The country where the server hardware is located.

Return type str

server_os
Get the operating system the server is running on.

Returns The name of the servers OS.

Return type str or NoneType

server_software
Get a list of the server's software stack.

Note: This does only work for localhosts, because most public servers don't list any software configuration in the HTTP response header.

Returns List of tuples containing both name and version for each software listed in the http header.

Return type list

title
Get the website's title.

The content of the first `<title>` tag in the HTML code is used.

Returns The title of the website.

Return type str

url
Get the website's URL.

Note: **This will not always return the URL you've passed to the `Info` constructor.** For example, the server may redirect to another page, and this function will return the URL of the website you was redirected to. If the website implements a client side redirect, you will not be redirected but get the URL you've passed before.

Example for clarification:

Using a fresh install of a recent *XAMPP*, `http://localhost` will redirect to `http://localhost/dashboard/`:

```
>>> import wsinfo
>>> w = wsinfo.Info("http://localhost")
>>> w.url
'http://localhost/dashboard/'
```

The original URL you've passed to the `Info` constructor is stored in the class attribute `_url`:

```
>>> w._url
'http://localhost'
```

Returns URL of the website.

Return type str

1.5 Changes

1.5.1 1.3.0

- Added properties: `content_type`, `http_header_dict` and `server_os`
- Correct handling of HTTP Errors (retrieve error page)
- Documentation updates
- Code cleanup
- Minor fixes and improvements

1.5.2 1.2.0

- Hosted docs on readthedocs.io
- Minor documentation changes

1.5.3 1.1.0

- Added function to list a websites heading structure
- Documentation improvements
- Code formatting
- Minor improvements
- Added/extended project infrastructure:
  - GitHub
  - PyPI
  - TravisCI
  - Landscape

1.5.4 1.0.0

- Initial release

1.6 License

The wsinfo source code is distributed under the terms of the MIT license, see below:

```
MIT License

Copyright (c) 2016 Linus Groh

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
```

```
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

CHAPTER 2

Indices and tables

- genindex
- modindex
- search

W

C

F

H

I

S

T

U

W