**Siteinfo\_web\_app.py Readme**

**Author: Kay Wang**

**Date: Oct.9 2017**

**Description:**

siteinfo\_web\_app.py is a Web application written in Python 3, it uses a lightweight pythonic HTTP framework - CherryPy which is a pure Python library.

Source code of the app:

siteinfo\_web\_app.py

siteinfo.py

app.conf

**How to run:**

siteinfo\_web\_app.py

**Configuration:**

App.conf:

[app]

server\_name = xxx.xxx.xxx.xxx # IP address of the HTTP server

server\_port = 80 # TCP port of the HTTP server

**Output:**

1. Title, Description, Keywords

Get the data by parsing HTML response for accessing the url.

Use Python library selenium in order to get the complete JavaScript rendered page.

Selenium requires a Web driver to interface with the chosen browser:

FireFox: geckodriver

Chrome: chromedriver

Safari: https://webkit.org/blog/6900/webdriver-support-in-safari-10/

2. Organization, Email, Telephone, Fax, Address

Get the data by calling whoisxmlapi RESTful API

3. Time Zone Offset, Time Zone Name

Call Google Map Geocoding RESTful API to get the latitude

and longitude from the address, then call Google Map Time

zone RESTful API to get the time zone information.

4. E-commerce Platform

Get the data by parsing HTML response for accessing www.builtwith.com.

Use Python library selenium in order to get the complete JavaScript rendered page.

In my opinion, the best way is calling a RESTful API to get the E-commerce platform

information, but I couldn't find it due to time limitation, or maybe it doesn't exist.

5. Alexa Global Ranking

Get the data by calling Alexa Web Information Service API.

I installed python-awis, however it didn't work because it was written in Python 2,

I changed the code to make it work in Python 3.

**Python Libraries and Web Drivers:**

BeautifulSoup - HTML file parser

Selenium Python Bindings - API to access Selenium WebDrivers

geckodriver - Web driver to interface with Firefox

python-awis - AWIS Python Wrapper

cherrypy - lightweight pythonic HTTP framework

The response format of the 3rd party RESTful API is XML, I use Python standard library ElementTree to parse

**Sample output:**



