
mapscookiegettercli Documentation

Release 0.2.0

Costas Tyfoxylos

Mar 20, 2021

Contents

1	mapscookiegettercli	3
1.1	Development Workflow	3
1.2	Important Information	4
1.3	Project Features	4
2	Installation	5
3	Usage	7
4	Contributing	9
4.1	Submit Feedback	9
5	mapscookiegettercli	11
5.1	mapscookiegettercli package	11
6	Credits	15
6.1	Development Lead	15
6.2	Contributors	15
7	History	17
8	0.0.1 (04-03-2019)	19
9	0.1.0 (25-03-2019)	21
10	0.1.1 (28-03-2019)	23
11	0.2.0 (16-06-2019)	25
12	Indices and tables	27
	Python Module Index	29
	Index	31

Contents:

- POSSIBLY DEPRECATED SINCE THERE ARE REPORTS OF GOOGLE BLOCKING THIS. SEE #1

A tool to retrieve the cookies from a google authentication process towards the google maps service to be used with locationsharinglib.

- Documentation: <https://mapscookiegettercli.readthedocs.org/en/latest>

1.1 Development Workflow

The workflow supports the following steps

- lint
- test
- build
- document
- upload
- graph

These actions are supported out of the box by the corresponding scripts under `_CI/scripts` directory with sane defaults based on best practices. Sourcing `setup_aliases.ps1` for windows powershell or `setup_aliases.sh` in bash on Mac or Linux will provide with handy aliases for the shell of all those commands prepended with an underscore.

The bootstrap script creates a `.venv` directory inside the project directory hosting the virtual environment. It uses `pipenv` for that. It is called by all other scripts before they do anything. So one could simply start by calling `_lint` and that would set up everything before it tried to actually lint the project

Once the code is ready to be delivered the `_tag` script should be called accepting one of three arguments, `patch`, `minor`, `major` following the semantic versioning scheme. So for the initial delivery one would call

```
$ _tag --minor
```

which would bump the version of the project to 0.1.0 tag it in git and do a push and also ask for the change and automatically update HISTORY.rst with the version and the change provided.

So the full workflow after git is initialized is:

- repeat as necessary (of course it could be test - code - lint :)) * code * lint * test
- commit and push
- develop more through the code-lint-test cycle
- tag (with the appropriate argument)
- build
- upload (if you want to host your package in pypi)
- document (of course this could be run at any point)

1.2 Important Information

This template is based on pipenv. In order to be compatible with requirements.txt so the actual created package can be used by any part of the existing python ecosystem some hacks were needed. So when building a package out of this **do not** simple call

```
$ python setup.py sdist bdist_egg
```

as this will produce an unusable artifact with files missing. Instead use the provided build and upload scripts that create all the necessary files in the artifact.

1.3 Project Features

- Successfully detects and uses Firefox and Chrome in Mac (latest version tested), Linux (Ubuntu 18.04 with budgie desktop tested) and Windows 7
- Todo:
 - Test Internet explorer and Edge on Windows and test windows 8 and 10.
 - Extend error handling for all environments.

CHAPTER 2

Installation

At the command line:

```
$ pip install mapscookiegettercli
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv mapscookiegettercli  
$ pip install mapscookiegettercli
```

Or, if you are using pipenv:

```
$ pipenv install mapscookiegettercli
```

Or, if you are using pipx:

```
$ pipx install mapscookiegettercli
```


CHAPTER 3

Usage

To develop on mapscookiegettercli:

```
# The following commands require pipenv as a dependency

# To lint the project
_CI/scripts/lint.py

# To execute the testing
_CI/scripts/test.py

# To create a graph of the package and dependency tree
_CI/scripts/graph.py

# To build a package of the project under the directory "dist/"
_CI/scripts/build.py

# To see the package version
_CI/scripts/tag.py

# To bump semantic versioning [--major|--minor|--patch]
_CI/scripts/tag.py --major|--minor|--patch

# To upload the project to a pypi repo if user and password are properly provided
_CI/scripts/upload.py

# To build the documentation of the project
_CI/scripts/document.py
```

To use mapscookiegettercli:

This tool is written for python3.7

Follow installation procedure from INSTALLATION.srt **for** your appropriate toolset.

(continues on next page)

(continued from previous page)

```
# execute the tool
maps-cookie-getter

# After the full login process the browser should be terminated and a "location_
↳sharing.cookies"
# file should be located at the same location that can be provided to the_
↳locationsharinglib.
```

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

4.1 Submit Feedback

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.

4.1.1 Get Started!

Ready to contribute? Here's how to set up *mapscookiegettercli* for local development. Using of *pipenv* is highly recommended.

1. Clone your fork locally:

```
$ git clone https://github.com/costastf/mapscookiegettercli
```

2. Install your local copy into a virtualenv. Assuming you have *pipenv* installed, this is how you set up your clone for local development:

```
$ cd mapscookiegettercli/  
$ pipenv install --ignore-pipfile
```

3. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally. Do your development while using the CI capabilities and making sure the code passes lint, test, build and document stages.

4. Commit your changes and push your branch to the server:

```
$ git add .  
$ git commit -m "Your detailed description of your changes."  
$ git push origin name-of-your-bugfix-or-feature
```

5. Submit a merge request

5.1 mapscookiegettercli package

5.1.1 Subpackages

mapscookiegettercli.browsers package

Submodules

mapscookiegettercli.browsers.chrome module

chrome package

class mapscookiegettercli.browsers.chrome.**Chrome**
Bases: object
Bootstraps a chrome selenium driver with the required settings

mapscookiegettercli.browsers.edge module

edge package

class mapscookiegettercli.browsers.edge.**Edge**
Bases: object
Bootstraps an edge selenium driver with the required settings

mapscookiegettercli.browsers.firefox module

firefox package

class mapscookiegettercli.browsers.firefox.**Firefox**
Bases: object
Bootstraps a firefox selenium driver with the required settings

mapscookiegettercli.browsers.ie module

ie package

class mapscookiegettercli.browsers.ie.**IE**
Bases: object
Bootstraps an internet explorer selenium driver with the required settings

Module contents

library package

Import all parts from library here

mapscookiegettercli.library package

Submodules

mapscookiegettercli.library.cookiegetter module

Main code for cookiegetter

class mapscookiegettercli.library.cookiegetter.**CookieGetter**
Bases: object
Object able to retrieve the cookies from an interactive login session to a google maps service
run (*cookie_file_name*='location_sharing.cookies')
Executes the process and saves the cookies
Parameters **cookie_file_name** (*str*) – The path and name of the exported cookie file
Returns:

Module contents

library package

Import all parts from library here

5.1.2 Submodules

5.1.3 mapscookiegettercli.mapscookiegettercli module

Main code for mapscookiegettercli

`mapscookiegettercli.mapscookiegettercli.get_arguments()`

Gets us the cli arguments.

Returns the args as parsed from the argparser.

`mapscookiegettercli.mapscookiegettercli.main()`

Main method.

This method holds what you want to execute when the script is run on command line.

5.1.4 mapscookiegettercli.mapscookiegettercliexceptions module

Custom exception code for mapscookiegettercli

exception `mapscookiegettercli.mapscookiegettercliexceptions.UnsupportedDefaultBrowser`

Bases: `Exception`

The browser could not be identified or is not supported.

exception `mapscookiegettercli.mapscookiegettercliexceptions.UnsupportedOS`

Bases: `Exception`

The os identified is not a supported one.

5.1.5 Module contents

mapscookiegettercli package

Import all parts from mapscookiegettercli here

6.1 Development Lead

- Costas Tyfoxylos <costas.tyf@gmail.com>

6.2 Contributors

- <https://github.com/Protonex>

CHAPTER 7

History

CHAPTER 8

0.0.1 (04-03-2019)

- First code creation

CHAPTER 9

0.1.0 (25-03-2019)

- Initial implementation working for firefox and chrome tested lightly on Mac, Ubuntu 18.04 and Windows 7

CHAPTER 10

0.1.1 (28-03-2019)

- Removed fstring instance so python3.7 is not a hard requirement

CHAPTER 11

0.2.0 (16-06-2019)

- Initial release, enforced en locale for worldwide compatibility contributed by “<https://github.com/Protonex>”

CHAPTER 12

Indices and tables

- `genindex`
- `modindex`
- `search`

m

- mapscookiegettercli, [13](#)
- mapscookiegettercli.browsers, [12](#)
- mapscookiegettercli.browsers.chrome, [11](#)
- mapscookiegettercli.browsers.edge, [11](#)
- mapscookiegettercli.browsers.firefox,
 [11](#)
- mapscookiegettercli.browsers.ie, [12](#)
- mapscookiegettercli.library, [12](#)
- mapscookiegettercli.library.cookiegetter,
 [12](#)
- mapscookiegettercli.mapscookiegettercli,
 [12](#)
- mapscookiegettercli.mapscookiegettercliexceptions,
 [13](#)

C

Chrome (*class in mapscookiegettercli.browsers.chrome*), 11

CookieGetter (*class in mapscookiegettercli.library.cookiegetter*), 12

E

Edge (*class in mapscookiegettercli.browsers.edge*), 11

F

Firefox (*class in mapscookiegettercli.browsers.firefox*), 11

G

get_arguments() (*in module mapscookiegettercli.mapscookiegettercli*), 12

I

IE (*class in mapscookiegettercli.browsers.ie*), 12

M

main() (*in module mapscookiegettercli.mapscookiegettercli*), 13

mapscookiegettercli (*module*), 13

mapscookiegettercli.browsers (*module*), 12

mapscookiegettercli.browsers.chrome (*module*), 11

mapscookiegettercli.browsers.edge (*module*), 11

mapscookiegettercli.browsers.firefox (*module*), 11

mapscookiegettercli.browsers.ie (*module*), 12

mapscookiegettercli.library (*module*), 12

mapscookiegettercli.library.cookiegetter (*module*), 12

mapscookiegettercli.mapscookiegettercli (*module*), 12

mapscookiegettercli.mapscookiegettercliexceptions (*module*), 13

R

run() (*mapscookiegettercli.library.cookiegetter.CookieGetter method*), 12

U

UnsupportedDefaultBrowser, 13

UnsupportedOS, 13