
DecryptLogin

Release 0.3.0

Zhenchao Jin

Aug 29, 2022

CONTENTS

1	Statements	1
2	Support List	3
3	Install	5
3.1	Environment	5
3.2	Dependency Package	5
3.3	Pip Install	5
3.4	Source Code Install	6
4	Quick Start	7
4.1	Login in a website with three lines of code	7
4.2	Login in a website with login.Client	7
4.3	Deal with captcha	7
4.4	Add proxies	8
4.5	Save cookies	8
4.6	Load cookies	8
5	Simulated Login	9
5.1	login.Login	9
5.2	login.Client	9
5.3	Supported Websites	10
5.4	Utility Functions	27
6	Practice with DecryptLogin	29
6.1	Install	29
6.2	Support List	29
6.3	Quick Start	29
7	Recommended Projects	31
8	About Me	33

STATEMENTS

This repo is created for learning python.

If I find that anyone leverage this project in an illegal way, I will delete this project immediately.

SUPPORT LIST

3.1 Environment

Here is the basic environment information:

- OS: Win10 / Mac OS / Linux
- Python: 3.6~3.8

3.2 Dependency Package

Dependencies requirement:

- rsa >= 4.0
- qrcode >= 6.1
- pillow >= 6.0.0
- requests >= 2.22.0
- pycryptodome >= 3.8.1
- requests_toolbelt >= 0.9.1
- gmssl >= 3.2.1
- PyExecJS >= 1.5.1 (the version of Node.js is v10.15.3 for my personal environment)

3.3 Pip Install

Run the following command in your terminal (Python should be in the develop environment):

```
pip install DecryptLogin
```

3.4 Source Code Install

1. Online

Run the following command in your terminal (Python and git should be in the develop environment):

```
pip install git+https://github.com/CharlesPikachu/DecryptLogin.git@master
```

2. Offline

First, you should clone the project in your computer:

```
git clone https://github.com/CharlesPikachu/DecryptLogin.git
```

Then, you should enter the project directory by running the following command:

```
cd DecryptLogin
```

Finally, you should run the following command in your terminal (Python should be in the develop environment) to install DecryptLogin:

```
python setup.py install
```

QUICK START

4.1 Login in a website with three lines of code

You can use the following three lines of code to easily implement a simulated login operation for any website in the support list. Take twitter as an example:

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.twitter(username='Your Username', password='Your Password')
```

where `infos_return` is a dict object, which contains some user information(e.g., `userid`) that may be useful. The `session` is a `requests.Session` object which has logged in the target website. Here is a screenshot:

4.2 Login in a website with `login.Client`

You can also login in the website by leveraging `login.Client`:

```
from DecryptLogin import login

client = login.Client()
weibo = client.weibo(reload_history=True)
infos_return, session = weibo.login('me', 'pass', mode='scanqr')
```

“`reload_history=True`” is used to reload the corresponding historical session saved in the computer and check whether the historical session is expired. If the historical session is expired, we will start a new login operation. Here is a screenshot:

4.3 Deal with captcha

By default, the users have to enter the captcha manually. If you want to deal with the captcha automatically, you can define a captcha identification function and pass it into the corresponding login api. Here is an example:

```
from PIL import Image
from DecryptLogin import login

"""the captcha identification function"""
def cracker(imagepath):
```

(continues on next page)

(continued from previous page)

```
# open captcha
img = Image.open(imagepath)
# identify captcha
result = IdentifyAPI(img)
# return the identification result
return result

lg = login.Login()
infos_return, session = lg.baidupan(username='Your Username', password='Your Password',
↳crack_captcha_func=cracker)
```

4.4 Add proxies

If you want to add proxies for the simulated login operation, you can pass the proxies into the corresponding login api as the following example:

```
from DecryptLogin import login

lg = login.Login()
proxies = {'https': '127.0.0.1:1080'}
infos_return, session = lg.bilibili(username='Your Username', password='Your Password',
↳proxies=proxies)
```

where the format of proxies is the same as `proxies` for requests.

4.5 Save cookies

You can save the session cookies as the following example:

```
from DecryptLogin.modules.utils.cookies import saveSessionCookies

session = requests.Session()
session.get(url)
saveSessionCookies(session=session, cookiespath='PATH to SAVE COOKIES (e.g., cookies.pkl)
↳')
```

4.6 Load cookies

You can load the cookies into a `requests.Session` as the following example:

```
from DecryptLogin.modules.utils.cookies import loadSessionCookies

session = requests.Session()
infos_return, session = loadSessionCookies(session=session, cookiespath='COOKIES PATH to
↳be LOADED')
```

SIMULATED LOGIN

5.1 login.Login

For users who only want to simply obtain the login-in session, it is recommended to use login.Login. Specifically, here is an example:

```
from DecryptLogin import login

# the instanced Login class object
lg = login.Login()
# use the provided api function to login in the target website (e.g., twitter)
infos_return, session = lg.twitter(username='Your Username', password='Your Password')
```

The api functions for logging in the corresponding websites all support the following arguments:

- username: the username for login in the target website,
- password: the password for login in the target website,
- mode: pc/mobile/scanqr, using the default setting is recommended,
- crack_captcha_func: a user-defined captcha identification function, the input of this function is the image path of captcha and it should return the recognition result of captcha,
- proxies: use proxies during the simulated login, the supported formats of proxies is the same as [Requests](#).

5.2 login.Client

login.Client leverages the instanced website client to perform logging operation. Specifically, the codes could be implemented as follows:

```
from DecryptLogin import login

# the instanced client
client = login.Client()
# the instanced weibo
weibo = client.weibo(reload_history=True)
# use the login function to login in weibo
infos_return, session = weibo.login('me', 'pass', 'scanqr')
```

The instanced websites all support the following arguments:

- reload_history: whether try to reload the corresponding historical session saved in the computer.

The login functions for logging in the corresponding websites all support the following arguments:

- username: the username for login in the target website,
- password: the password for login in the target website,
- mode: pc/mobile/scanqr, using the default setting is recommended,
- crack_captcha_func: a user-defined captcha identification function, the input of this function is the image path of captcha and it should return the recognition result of captcha,
- proxies: use proxies during the simulated login, the supported formats of proxies is the same as [Requests](#).

5.3 Supported Websites

5.3.1 weibo

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.weibo(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported since the sms code is always required when you want to login in PC Mode.

2.Mobile Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.weibo(username, password, 'mobile')
```

The user-defined crack_captcha_func has not been supported since the sms code is always required when you want to login in Mobile Mode.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.weibo('', '', 'scanqr')
```

Then, you can leverage the APP of weibo to scan the qr code to login in the website.

5.3.2 douban

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.douban(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.douban('', '', 'scanqr')
```

Then, you can leverage the APP of douban to scan the qr code to login in the website.

5.3.3 github

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.github(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.4 music163

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.music163(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.music163('', '', 'scanqr')
```

Then, you can leverage the APP of music163 to scan the qr code to login in the website.

5.3.5 zt12306

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.zt12306(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported since the sms code is always required when you want to login in PC Mode.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.zt12306('', '', 'scanqr')
```

Then, you can leverage the APP of zt12306 to scan the qr code to login in the website.

5.3.6 QQZone

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.QQZone('', '', 'scanqr')
```

Then, you can leverage the APP of TIM or QQ to scan the qr code to login in the website.

5.3.7 QQQun

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.QQQun('', '', 'scanqr')
```

Then, you can leverage the APP of TIM or QQ to scan the qr code to login in the website.

5.3.8 QQId

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.QQId('', '', 'scanqr')
```

Then, you can leverage the APP of TIM or QQ to scan the qr code to login in the website.

5.3.9 zhihu

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.zhihu(username, password, 'pc')
```

The crack_captcha_func can be defined as follow:

```
def cracker(imagepath):
    return 'LOVE'
```

The mail is recommended to be as the username.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.zhihu('', '', 'scanqr')
```

Then, you can leverage the APP of zhihu to scan the qr code to login in the website.

5.3.10 bilibili

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.bilibili(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported since the sms code is always required when you want to login in PC Mode.

2.Mobile Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.bilibili(username, password, 'mobile')
```

The user-defined crack_captcha_func has not been supported since the sms code is always required when you want to login in Mobile Mode.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.bilibili('', '', 'scanqr')
```

Then, you can leverage the APP of bilibili to scan the qr code to login in the website.

5.3.11 toutiao

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.toutiao('', '', 'scanqr')
```

Then, you can leverage the APP of toutiao to scan the qr code to login in the website.

5.3.12 taobao

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.taobao('', '', 'scanqr')
```

Then, you can leverage the APP of taobao to scan the qr code to login in the website.

5.3.13 jingdong

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.jingdong('', '', 'scanqr')
```

Then, you can leverage the APP of jingdong to scan the qr code to login in the website.

5.3.14 ifeng

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.ifeng(username, password, 'pc')
```

The crack_captcha_func can be defined as follow:

```
def cracker(imagepath):
    return 'LOVE'
```

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.15 sohu

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.sohu(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.sohu(username, password, 'mobile')
```

The user-defined crack_captcha_func has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.16 zgconline

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.zgconline(username, password, 'pc')
```

The crack_captcha_func can be defined as follow:

```
def cracker(imagepath):
    return 'LOVE'
```

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.17 lagou

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.lagou(username, password, 'pc')
```

The crack_captcha_func can be defined as follow:

```
def cracker(imagepath):
    return 'LOVE'
```

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.18 twitter

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.twitter(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.twitter(username, password, 'mobile')
```

The user-defined crack_captcha_func has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.19 eSurfing

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.eSurfing('', '', 'scanqr')
```

Then, you can leverage the APP of eSurfing to scan the qr code to login in the website.

5.3.20 renren

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.renren(username, password, 'pc')
```

The crack_captcha_func can be defined as follow:

```
def cracker(imagepath):
    return 'LOVE'
```

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.21 w3cschool

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.w3cschool(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.22 fishc

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.fishc(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.23 youdao

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.youdao(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.24 baidupan

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.baidupan(username, password, 'pc')
```

The crack_captcha_func can be defined as follow:

```
def cracker(imagepath):
    return 'LOVE'
```

The sms code is always required for secondary verification when you want to login in PC Mode due to the security mechanism.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.25 stackoverflow

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.stackoverflow(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.26 codalab

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.codalab(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.27 pypi

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.pypi(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.28 douyu

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.douyu('', '', 'scanqr')
```

Then, you can leverage the APP of douyu, TIM or QQ to scan the qr code to login in the website.

5.3.29 migu

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.migu(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.30 qunar

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.qunar(username, password, 'pc')
```

The crack_captcha_func can be defined as follow:

```
def cracker(imagepath):  
    return 'LOVE'
```

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.31 mieshop

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login  
  
lg = login.Login()  
infos_return, session = lg.mieshop(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.32 mpweixin

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login  
  
lg = login.Login()  
infos_return, session = lg.mpweixin(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported since the qr code scanned by wechat is always required for the secondary verification when you want to login in PC Mode.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.33 baidutieba

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.baidutieba('', '', 'scanqr')
```

Then, you can leverage the APP of baidutieba to scan the qr code to login in the website.

5.3.34 dazhongdianping

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.dazhongdianping('', '', 'scanqr')
```

Then, you can leverage the APP of dazhongdianping to scan the qr code to login in the website.

5.3.35 jianguoyun

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.jianguoyun(username, password, 'pc')
```

The user-defined crack_captcha_func has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.36 cloud189

1.PC Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.cloud189(username, password, 'pc')
```

The crack_captcha_func can be defined as follow:

```
def cracker(imagepath):
    return 'LOVE'
```

2.Mobile Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.cloud189(username, password, 'mobile')
```

The user-defined crack_captcha_func has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.37 qqmusic

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.qqmusic('', '', 'scanqr')
```

Then, you can leverage the APP of QQ or TIM to scan the qr code to login in the website.

5.3.38 ximalaya

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.ximalaya('', '', 'scanqr')
```

Then, you can leverage the APP of ximalaya to scan the qr code to login in the website.

5.3.39 icourse163

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.icourse163(username, password, 'mobile')
```

The user-defined crack_captcha_func has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.40 xiaomihealth

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.xiaomihealth(username, password, 'mobile')
```

The user-defined crack_captcha_func has not been supported.

3.Scanqr Mode

The scanqr mode has not been supported.

5.3.41 tencentvideo

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.tencentvideo('', '', 'scanqr')
```

Then, you can leverage the APP of QQ or TIM to scan the qr code to login in the website.

5.3.42 baidu

1.PC Mode

The pc mode has not been supported.

2.Mobile Mode

The mobile mode has not been supported.

3.Scanqr Mode

The sample codes is as follow::

```
from DecryptLogin import login

lg = login.Login()
infos_return, session = lg.baidu('', '', 'scanqr')
```

Then, you can leverage the APP of baidu to scan the qr code to login in the website.

5.4 Utility Functions

5.4.1 Cookies

1.Save cookies

You can save the session cookies as the following example:

```
from DecryptLogin.modules.utils.cookies import saveSessionCookies

session = requests.Session()
session.get(url)
infos_return = saveSessionCookies(session=session, cookiespath='PATH to SAVE COOKIES (e.
↳g., cookies.pkl)')
```

The explanation of the arguments:

```
Function:
    save the cookies in requests.Session
Input:
    --session: the requests.Session object
    --cookiespath: the file path to save cookies
    --encoding: the encoding of the file for saving cookies
Return:
    --infos_return: return the flag of whether save the cookies successfully, if
    ↪ fail to save, also return the detailed error information
```

2.Load cookies

You can load the cookies into a requests.Session as the following example:

```
from DecryptLogin.modules.utils.cookies import loadSessionCookies

session = requests.Session()
infos_return, session = loadSessionCookies(session=session, cookiespath='COOKIES PATH to
    ↪ be LOADED')
```

The explanation of the arguments:

```
Function:
    load the cookies into requests.Session
Input:
    --session: the requests.Session object before loading cookies
    --cookiespath: the file path of the saved cookies
    --encoding: the encoding of the cookies file
Return:
    --infos_return: return the flag of whether load the cookies successfully, if
    ↪ fail to load, also return the detailed error information
    --session: the requests.Session object after loading cookies
```


PRACTICE WITH DECRYPTLOGIN

6.1 Install

Preparation

- `ffmpeg`: You should set `ffmpeg` in environment variable.
- `aria2c`: You should set `aria2c` in environment variable.

Pip install

```
run "pip install DecryptLoginExamples"
```

6.2 Support List

6.3 Quick Start

```
from DecryptLoginExamples import client

config = {
    'username': 'charlespikachu',
    'time_interval': 1800,
}
crawler_executor = client.Client()
crawler_executor.executor('bilibililottery', config=config)
```


RECOMMENDED PROJECTS

- **Games:** Create interesting games by pure python.
- **DecryptLogin:** APIs for logging some websites by using requests.
- **Musicdl:** A lightweight music downloader written by pure python.
- **Videodl:** A lightweight video downloader written by pure python.
- **Pytools:** Some useful tools written by pure python.
- **PikachuWeChat:** Play WeChat with itchat-uos.
- **SSSegmentation:** An Open Source Strongly Supervised Semantic Segmentation Toolbox Based on PyTorch.
- **Pydrawing:** Beautify your image or video.
- **ImageCompressor:** Image compressors written by pure python.
- **FreeProxy:** Collecting free proxies from internet.
- **Constellation:** Beautiful starry sky written by js.
- **Paperdl:** Search and download paper from specific websites.
- **Sciogovterminal:** Browse “The State Council Information Office of the People’s Republic of China” in the terminal.
- **CodeFree:** Make no code a reality.
- **DeepLearningToys:** Some deep learning toys implemented in pytorch.
- **DataAnalysis:** Some data analysis projects in charles_pikachu.
- **Imagedl:** Search and download images from specific websites.
- **Pytoydl:** A toy deep learning framework built upon numpy.
- **NovelDL:** Search and download novels from some specific websites.

ABOUT ME

I'm a student whose research interests include computer vision and information security.

WeChat public account: Charles_pikachu (*i.e.*, Charles)

Github: <https://github.com/CharlesPikachu>

Zhihu: https://www.zhihu.com/people/charles_pikachu

Bilibili: <https://space.bilibili.com/406756145>

Email: charlesblwx@gmail.com