



链滴

# alexamaster 赚美刀之刷 Good Votes

作者: [alanfans](#)

原文链接: <https://ld246.com/article/1557112475974>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

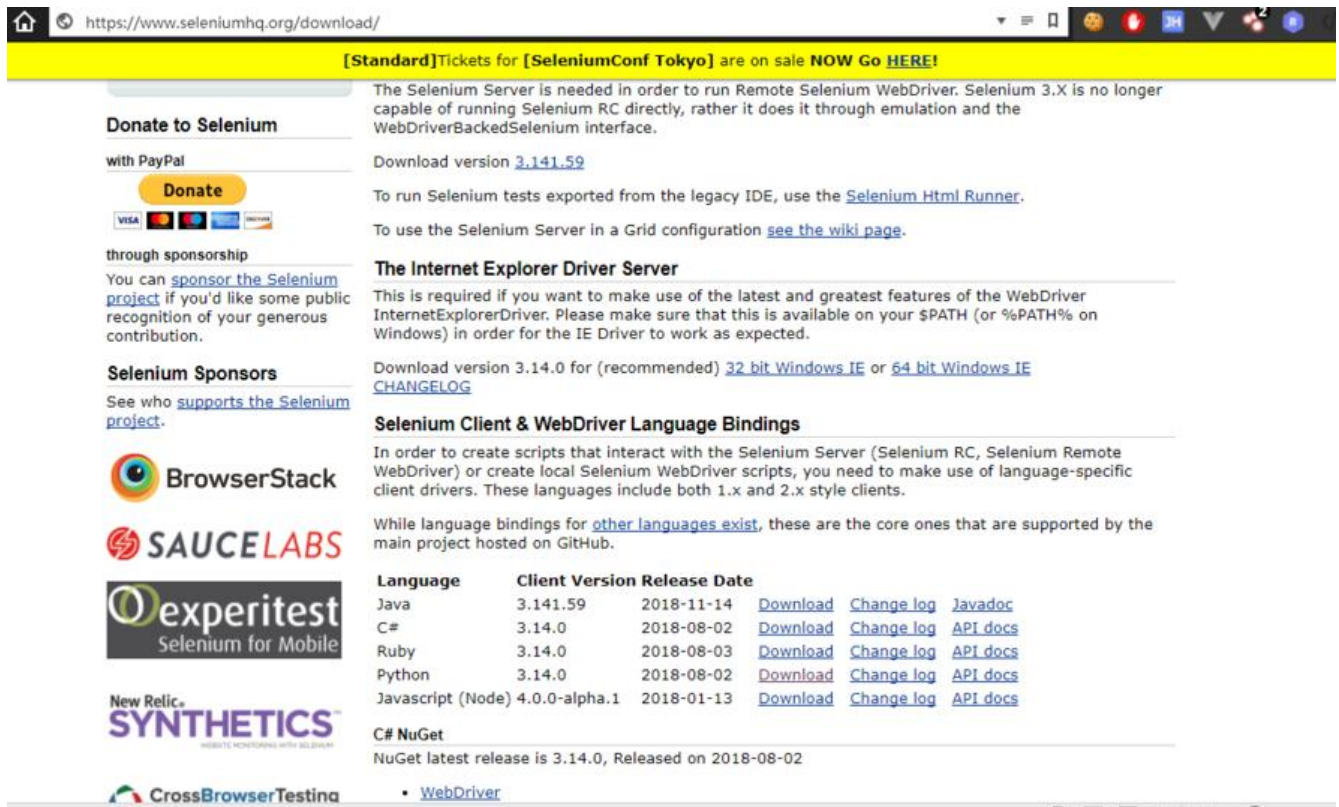
工具: python,selenium

alexamaster赚美刀提现需要Good Votes(gv), 所需gv跟据提现次数呈指数级增长,人工手动点击已可能, 所以用自动化测试工具

## 下载

安装selenium [pip install selenium](#)

下载驱动包, 我选用的python版,



**[Standard] Tickets for [SeleniumConf Tokyo] are on sale NOW Go [HERE!](#)**

The Selenium Server is needed in order to run Remote Selenium WebDriver. Selenium 3.X is no longer capable of running Selenium RC directly, rather it does it through emulation and the WebDriverBackedSelenium interface.

Download version [3.141.59](#)

To run Selenium tests exported from the legacy IDE, use the [Selenium Html Runner](#).

To use the Selenium Server in a Grid configuration [see the wiki page](#).

### The Internet Explorer Driver Server

This is required if you want to make use of the latest and greatest features of the WebDriver InternetExplorerDriver. Please make sure that this is available on your \$PATH (or %PATH% on Windows) in order for the IE Driver to work as expected.

Download version 3.14.0 for (recommended) [32 bit Windows IE](#) or [64 bit Windows IE](#)  
[CHANGELOG](#)

### Selenium Client & WebDriver Language Bindings

In order to create scripts that interact with the Selenium Server (Selenium RC, Selenium Remote WebDriver) or create local Selenium WebDriver scripts, you need to make use of language-specific client drivers. These languages include both 1.x and 2.x style clients.

While language bindings for [other languages exist](#), these are the core ones that are supported by the main project hosted on GitHub.

Language	Client Version	Release Date	Download	Change log	Javadoc
Java	3.141.59	2018-11-14	<a href="#">Download</a>	<a href="#">Change log</a>	<a href="#">Javadoc</a>
C#	3.14.0	2018-08-02	<a href="#">Download</a>	<a href="#">Change log</a>	<a href="#">API docs</a>
Ruby	3.14.0	2018-08-03	<a href="#">Download</a>	<a href="#">Change log</a>	<a href="#">API docs</a>
Python	3.14.0	2018-08-02	<a href="#">Download</a>	<a href="#">Change log</a>	<a href="#">API docs</a>
Javascript (Node)	4.0.0-alpha.1	2018-01-13	<a href="#">Download</a>	<a href="#">Change log</a>	<a href="#">API docs</a>

### C# NuGet

NuGet latest release is 3.14.0, Released on 2018-08-02

- [WebDriver](#)

以chrome为例

https://pypi.org/project/selenium/

lmtierney /usr/bin or /usr/local/bin.

lsemmerau Failure to observe this step will give you an error `selenium.common.exceptions.WebDriverException: Message: 'geckodriver' executable needs to be in PATH.`

maikroeder Other supported browsers will have their own drivers available. Links to some of the more popular browser drivers follow.

tebeka

Chrome:	<a href="https://sites.google.com/a/chromium.org/chromedriver/downloads">https://sites.google.com/a/chromium.org/chromedriver/downloads</a>
Edge:	<a href="https://developer.microsoft.com/en-us/microsoft-edge/tools/webdriver/">https://developer.microsoft.com/en-us/microsoft-edge/tools/webdriver/</a>
Firefox:	<a href="https://github.com/mozilla/geckodriver/releases">https://github.com/mozilla/geckodriver/releases</a>
Safari:	<a href="https://webkit.org/blog/6900/webdriver-support-in-safari-10/">https://webkit.org/blog/6900/webdriver-support-in-safari-10/</a>

**Classifiers**

**Development Status**  
5 - Production/Stable

**Intended Audience**  
Developers

**License**  
OSI Approved :: Apache Software License

**Operating System**  
MacOS :: MacOS X  
Microsoft :: Windows  
POSIX

**Programming Language**  
Python

**Example 0:**

- open a new Firefox browser
- load the page at the given URL

```
from selenium import webdriver

browser = webdriver.Firefox()
browser.get('http://seleniumhq.org')
```

## 版本对应

Google Chrome

Google Chrome 已是最新版本  
版本 74.0.3729.131 (正式版本) (64 位)

获取有关 Chrome 的帮助

## 代码

```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.action_chains import ActionChains
import time
class Am_static():
    account = {
        'email': '用户名',
        'passwd': '密码',
    }
    em_botton = {
        'earn': 'earn_points.php',
        'mem_pts': 'mem_pts',
        'ads': (By.XPATH, "//a[contains(@onclick, 'openSite')]"),
    }
    em_url = {
        'chrome': 'http://chromedriver.storage.googleapis.com/index.html',
```

```

'login' : 'https://www.alexamaster.net/sec/login.php',
'loginConfirm' : 'https://www.alexamaster.net/sec/loginconfirm.php',
'a' : 'https://www.alexamaster.net/a',
'profile' : 'https://www.alexamaster.net/a/my_profile.php',
'earn' : 'https://www.alexamaster.net/a/earn_points.php'
}

def __init__(self,browerType):
    if 'chrome' == browerType:
        self.brower = webdriver.Chrome('E:\Program Files\selenium\chromedriver.exe')
    else:
        self.brower = webdriver.Firefox(executable_path='E:\Program Files\selenium\geckodri
er.exe')

def login(self):
    self.brower.get(self.em_url.get('login'))
    # print("firefox:" + fbrower.page_source)
    try:
        _input = self.brower.execute_script("return $('form').find('input')")
        # email
        _input[0].send_keys(self.account.get('email'))
        # pwd
        _input[1].send_keys(self.account.get('passwd'))
        _input[2].click()
        time.sleep(3)
        self.brower.get(self.em_url.get('loginConfirm'))
        self.brower.get(self.em_url.get('a'))
        time.sleep(3)
        mem_pts = self.brower.find_element_by_id(self.em_botton.get('mem_pts'))
        print('mem_pts:'+mem_pts.text)
    except BaseException as identifier:
        self.login()

def visite_earn_page_scroll(self):
    try:
        # not find earn_pgae
        # button_earn = brower.find_element(By.LINK_TEXT,self.em_url.get("earn"))
        # button_earn.click()
        # brower.get(self.em_url.get('earn'))5,261
        self.brower.get(self.em_url.get('earn'))
        js = 'document.getElementsByClassName("main-panel")[0].scrollTop'
        height = self.brower.execute_script("return "+js)
        self.brower.execute_script(js+"="+str(height+10000))
        print('height:'+str(height+10000))
    except BaseException as identifier:
        # self.brower.refresh()
        self.visite_earn_page_scroll()

def checkGV(self):
    time.sleep(6)
    gvs_div = self.brower.execute_script("return $('timeline-panel')")
    hasGV = 0
    for gv in gvs_div:
        if '5 Good Votes from' in gv.text:

```

```

try:
    ads = gv.find_element(*self.em_botton.get('ads'))
    self.getGV(ads)
    hasGV = 1
except BaseException as identifier:
    print("error")

if hasGV == 0:
    self.visite_earn_page_scroll()
    self.checkGV()

def getGV(self,ads):
    ads.click()
    time.sleep(21)
    # switch_tab
    windows = self.brower.window_handles
    self.brower.switch_to.window(windows[1])
    self.brower.close()
    self.brower.switch_to.window(windows[0])
    goodconfirm = self.brower.find_element_by_class_name("swal2-confirm")
    goodconfirm.click()
    goodconfirm = self.brower.find_element_by_class_name("swal2-confirm")
    goodconfirm.click()
    time.sleep(3)
    self.checkGV()

test = Am_static('chrome')
test.login()
test.visite_earn_page_scroll()
test.checkGV()

```

## 效果图

```
1 You, a few seconds ago | 2 authors (You and others)
2 from selenium import webdriver
3 from selenium.webdriver.common.by import By
```

The screenshot shows a Selenium WebDriver browser window with the URL `https://proscooters.club`. The page content includes a search bar, a navigation menu with items like 'Pro Scooter shop', 'Categories', 'Contact Us', 'Terms & Conditions', and 'All Products', and a main section titled 'Pro Scooters'. The main text describes the benefits of pro scooters, such as their quality and maneuverability. A sidebar on the left contains social media icons and a newsletter sign-up form. The browser's developer tools are visible at the bottom, showing a list of elements with a height of 10000.