

第 5 章 Selenium 3(WebDriver)的安装

本章主要讲解 WebDriver 的安装与配置方法,请读者按照本章节的内容安装和配置好 WebDriver 的运行环境,以便在后续章节中,讲解基于 WebDriver 实例时,能及时执行及查看结果。

本书以笔者撰稿时最新的 Selenium(3.0.2)版本为例,介绍其安装及使用方法。

5.1 在 Python 中安装 WebDriver

(1) 安装好 Python、pip 工具后,尝试直接在 CMD 下输入 `pip install selenium`,如图 5-1 所示,如果成功直接跳转到第(5)步开始执行,笔者选择的是 `selenium==3.14.0` 版本。

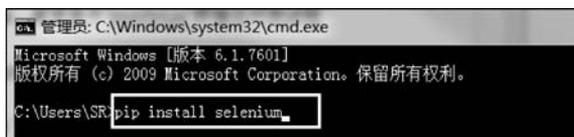


图 5-1

(2) 下载 Selenium 离线安装包,访问网址: <https://pypi.python.org/pypi/selenium>,选择扩展名为 `gz` 的源码包进行下载,如图 5-2 所示。

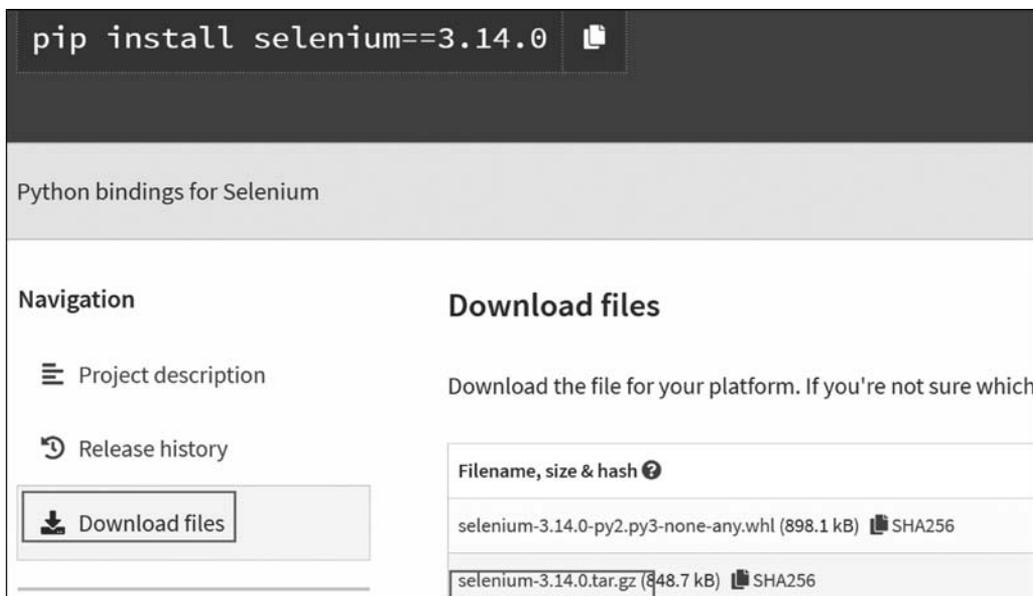


图 5-2



(3) 下载完以后会得到一个 selenium-3.14.0.tar.gz 离线安装文件,解压该文件到任意目录。CMD 下通过 cd 命令将当前的工作目录切换到 setup.py 文件所在的目录,如图 5-3 所示。

```
D:\>cd D:\selenium-3.14.0
D:\selenium-3.14.0>dir
驱动器 D 中的卷是 Study
卷的序列号是 6A05-836C

D:\selenium-3.14.0 的目录
2018/12/22  15:57    <DIR>          .
2018/12/22  15:57    <DIR>          ..
2018/08/02  21:06           25,201  CHANGES
2017/03/08  20:00           915   MANIFEST.in
2018/08/02  21:22           7,737  PKG-INFO
2018/07/26  17:59           5,650  README.rst
2018/12/22  15:57    <DIR>          selenium
2018/12/22  15:57    <DIR>          selenium.egg-info
2018/08/02  21:22           209   setup.cfg
2018/08/02  21:18           3,220  setup.py
                6 个文件          42,932 字节
                4 个目录    85,147,713,536 可用字节

D:\selenium-3.14.0>
```

图 5-3

(4) 然后在 CMD 下执行 python setup.py install 命令进行安装,安装成功后的界面如图 5-4 所示。

```
Finished processing dependencies for selenium==3.14.0
D:\selenium-3.14.0>
```

图 5-4

(5) 安装成功后,CMD 下输入 python 回车,进入 Python 交互模式,执行 import selenium 命令,如果没有报错,说明 Selenium 已经安装成功,如图 5-5 所示。

```
C:\Python\Python35\python35.exe
Python 3.5.4 (v3.5.4:3f56838, Aug 8 2017, 02:17:05) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import selenium
>>> ← 没有报错,说明安装成功了
```

图 5-5

5.2 第一个 WebDriver 脚本

测试目标:

使用 Firefox 浏览器验证 WebDriver 是否可用。

测试用例步骤:

- (1) 在 Firefox 浏览器中打开搜狗浏览器首页。
- (2) 在搜索输入框中输入“光荣之路自动化测试”。
- (3) 单击“搜索”按钮。
- (4) 页面显示搜索结果。

环境准备：

(1) 使用 Firefox 浏览器执行 Selenium 3 编写的自动化测试脚本时, 需要从 <https://github.com/mozilla/geckodriver/releases> 网址根据读者操作系统类型及浏览器位数(64 位驱动兼容 32 位 Firefox 浏览器) 下载对应的 WebDriver 操作 Firefox 浏览器的驱动程序, 如图 5-6 所示。

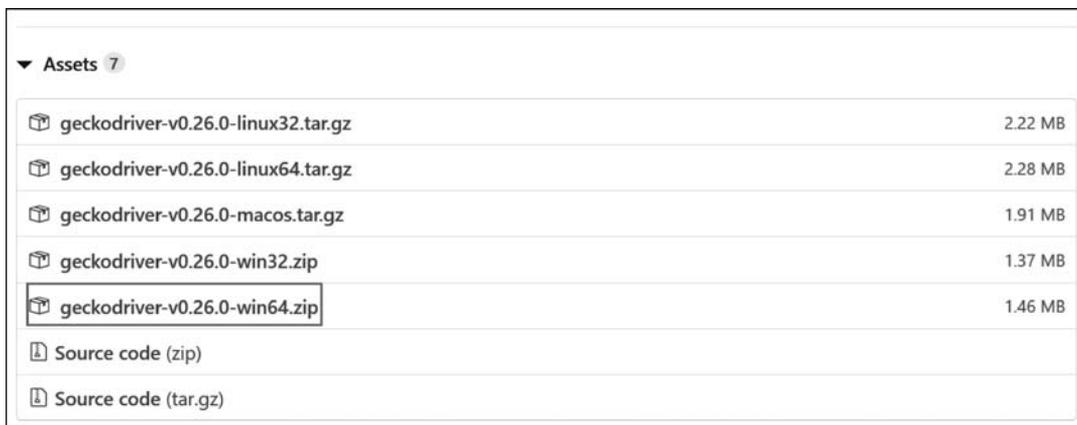


图 5-6

(2) 下载并解压得到 geckodriver.exe 文件, 将该文件保存到本地硬盘任意位置, 比如 D:\下。

测试脚本程序：

```
from selenium import webdriver
import time

# 通过 executable_path 参数指明 Firefox 驱动文件所在路径
driver = webdriver.Firefox(executable_path = "c:\\\\geckodriver")
# driver = webdriver.Chrome(executable_path = "c:\\\\chromedriver")
# 打开搜狗首页
driver.get("http://www.sogou.com")
# 清空搜索输入框默认内容
driver.find_element_by_id("query").clear()
# 在搜索输入框中输入"光荣之路自动化测试"
driver.find_element_by_id("query").send_keys("光荣之路自动化测试")
# 单击"搜索"按钮
driver.find_element_by_id("stb").click()
# 等待 3 秒
time.sleep(3)
# 退出浏览器
driver.quit()
```

PyCharm 中执行该脚本, 会看到程序自动启动浏览器, 访问搜狗首页, 并在搜索输入框中输入搜索关键内容“光荣之路自动化测试”, 单击搜索按钮后展示搜索结果, 3 秒后自动退出浏览器。



更多说明：

从 Selenium 3 版本开始,webdriver/firefox/webdriver.py 程序文件中的__init__.py 文件中,设置 executable_path="geckodriver",而 Selenium 2 是 executable_path="wires",所以使用 Selenium 3 编写的自动化测试脚本,使用 Firefox 浏览器测试时需要指明 Firefox 浏览器驱动程序 geckodriver.exe 文件所在路径。

问题及建议：

建议 Firefox 浏览器在安装时使用默认安装路径。如果使用了自定义安装路径,可能无法找到 Firefox.exe 文件来启动执行此测试脚本的 Firefox 浏览器。

报错 1：

```
Exception in thread "main" org.openqa.selenium.WebDriverException: Cannot find firefox binary in PATH. Make sure firefox is installed.
```

解决办法：

在 driver = webdriver.Firefox(executable_path = "c:\geckodriver")这行代码前一行增加如下代码：

```
os.environ["webdriver.firefox.driver"] = "C:\Program Files (x86)\Mozilla Firefox\firefox.exe"
```

其中"C:\ProgramFiles (x86)\Mozilla Firefox\firefox.exe"代表 firefox.exe 文件所在的路径,读者须根据自己机器上 firefox.exe 文件所在的路径进行修改。

报错 2：

```
WebDriverException: Message: Expected browser binary location, but unable to find binary in default location, no 'moz:firefoxOptions.binary' capability provided, and no binary flag set on the command line
```

解决方法：

脚本顶部导入 FirefoxBinary 模块：

```
from selenium.webdriver.firefox.firefox_binary import FirefoxBinary
```

修改启动浏览器的代码如下：

```
binary = FirefoxBinary('D:\\FirefoxPortable\\Firefox.exe')
driver = webdriver.Firefox(firefox_binary = binary,
    executable_path = r"c:\geckodriver")
```

5.3 各浏览器驱动的使用方法

Selenium 3 版本开始不再提供默认浏览器支持,所有浏览器都是通过各个浏览器官方提供相应的浏览器驱动进行支持,这使得运行在各种浏览器上的自动化测试更稳定。

除了本章 5.2 小节中直接通过 executable_path 参数指明支持对应浏览器的驱动程序文件方法外,读者还可以创建一个目录(比如 D:\driver\目录),把不同浏览器的驱动文件均放到该目录中(比如: geckodriver.exe、chromedriver.exe、MicrosoftWebDriver.exe、



IEDriverServer.exe、operadriver.exe 等),然后将该目录(比如 D:\driver\目录)添加到系统环境变量 path 中,WebDriver 在启动浏览器时,会自动到环境变量中设定的路径中寻找相应的驱动文件。



在本书中,笔者将采用 5.2 小节实例程序中介绍的添加浏览器驱动的方式,仅是为了更灵活。