

使用Python和Selenium进行Web自动化测试的实战指南

# Selenium自动化测试

## ——基于Python语言

Learning Selenium Testing  
Tools with Python

[印度] 昂米沙·冈迪察 (Unmesh Gundecha) 著

金鑫 熊志男 译

testwo测试窝 审



# Selenium自动化测试

## ——基于Python语言

[印度] 昂米沙·冈迪察 (Unmesh Gundecha) 著

金鑫 熊志男 译

testwo测试窝 审

人民邮电出版社

北京

## 图书在版编目 (C I P) 数据

Selenium自动化测试：基于 Python 语言 / (印)  
冈迪察·U (Unmesh Gundecha) 著；金鑫，熊志男译  
— 北京：人民邮电出版社，2018.1 (2018.8 重印)  
ISBN 978-7-115-46174-2

I. ①S… II. ①冈… ②金… ③熊… III. ①软件工  
具—自动检测 IV. ①TP311.561

中国版本图书馆CIP数据核字(2017)第274665号

## 版权声明

Copyright © Packt Publishing 2014. First published in the English language under the title“Learning Selenium Testing Tools with Python”, ISBN 978-1-78398-350-6.

All rights reserved.

本书中文简体字版由 Packt Publishing 公司授权人民邮电出版社出版。未经出版者书面许可，对本书的任何部分不得以任何方式或任何手段复制和传播。

版权所有，侵权必究。

- 
- ◆ 著 [印度] 昂米沙·冈迪察 (Unmesh Gundecha)
  - 译 金鑫 熊志男
  - 审 Testwo 测试窝
  - 责任编辑 张涛
  - 责任印制 焦志炜
  
  - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号  
邮编 100164 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京中石油彩色印刷有限责任公司印刷
  
  - ◆ 开本：800×1000 1/16  
印张：12.75  
字数：223 千字 2018 年 1 月第 1 版  
印数：3 001-3 500 册 2018 年 8 月北京第 2 次印刷  
著作权合同登记号 图字：01-2016-3957 号
- 

定价：49.00 元

读者服务热线：(010) 81055410 印装质量热线：(010) 81055316

反盗版热线：(010) 81055315

广告经营许可证：京东工商广登字 20170147 号

# 内容提要

Selenium 是一个主要用于 Web 应用程序自动化测试的工具集合，在行业内已经得到广泛的应用。本书介绍了如何用 Python 语言调用 Selenium WebDriver 接口进行自动化测试。主要内容为：基于 Python 的 Selenium WebDriver 入门知识、第一个 Selenium Python 脚本、使用 unittest 编写单元测试、生成 HTML 格式的测试报告、元素定位、Selenium Python API 介绍、元素等待机制、跨浏览器测试、移动端测试、编写一个 iOS 测试脚本、编写一个 Android 测试脚本、Page Object 与数据驱动测试、Selenium WebDriver 的高级特性、第三方工具与框架集成等核心技术。

本书适合任何软件测试人员阅读，也适合作为大专院校师生的学习用书和培训学校的教材。

# 推荐序

认识熊志男是在中国质量大会 BQConf 的活动上，交谈间很快就被志男对于测试领域的见解和趋势展望所折服，而这不仅仅是因为他有丰富的测试经验，而且还因为他作为测试窝的联合创始人，对国内外测试行业有深入的了解密不可分。本书正是一个例证。

尽管国内测试行业宣扬自动化测试已经很多年了，但是我们很难在自动化测试领域招聘到经验丰富的工程师，这说明自动化测试并没有成为国内测试领域的主流。

是因为测试人员不够努力导致的吗？我并不这么认为。

很多公司对于测试的投入是希望知道产品有多少缺陷、能否按时上线，所以相应的测试人员的工作都聚焦于如何高效地编写和执行测试用例，而自动化测试并不是第一选择。因为在线上压力巨大的情况下，如果不能评估出自动化测试的投入产出比，很难让项目经理在自动化测试上进行投入。所以倘若有实践自动化测试的想法，往往需要测试人员付出自己的时间来熟悉框架、编写和维护自动化测试。而如果不是事先对测试框架有了比较深入的认识，依靠自身自发进行自动化测试，并不会带来效率的明显提升。这样不仅会让项目经理失去信心，恐怕测试人员自己也心存疑虑了。

如果单从技术上考虑，究竟是什么阻碍着测试人员广泛使用自动化测试呢？首先，如果没有一个通用的测试框架，那么每做一个项目，测试人员就得学习一套新的工具 / 框架，这样的学习成本太高了。在工期很紧的情况下更是如此。其次，自动化测试的编写实际上是进行编码，如果使用 Java 和 C# 这些编程语言编写自动化测试，由于测试人员很难全面掌握这些语言的开发技巧，容易导致编写出的自动化测试代码比产品代码出现更多的缺陷。



本书直击这两方面，为测试人员解除了后顾之忧。

(1) Selenium WebDriver 作为业界通用的测试框架，不仅是 Web 测试的标准，在移动测试领域也是底层的核心驱动框架。所以掌握了 Selenium WebDriver，可以让我们在为 Web 产品和移动产品编写自动化测试时游刃有余。

(2) Python 作为动态语言，简化了严格的编程语法，使测试人员更容易掌握。同时 Python 也提供了丰富的 API 和扩展，测试人员可以很便利地调用或者集成其他语言编写的程序和类库，提高编写自动化测试的效率。

本书在讲述自动化测试编写的同时，结合业界主流的自动化测试开发模式，向读者介绍了多种测试相关知识（如 BDD 和持续集成）。非常推荐对测试有激情，希望快速提升自动化测试能力的朋友阅读本书。

黄 勇

现任 ThoughtWorks 中国区 QA Lead

# 译者序

起初接到本书的翻译邀约时，内心还是有一些困惑的。针对软件测试行业，特别是基于 Web 自动化测试领域，Selenium 已经是广泛使用的工具之一了，而且已被诸多测试同行认可并使用。为此，我们查阅了国内大量相关书籍或文章，发现当前 Selenium 的初学门槛其实并不高，测试工程师具备有功能测试经验，加之对 Web 前端技术的一定程度的理解，外加较熟练地掌握一门脚本语言，经过一段时间的项目锤炼，都能基本应对日常自动化测试任务。不过与此同时，我们也发现很多初学者遇到的诸多困惑，又或者在深入学习过程中难以克服的瓶颈。

行业内，能系统性介绍 WebDriver 原理、多类型 Server 运行方式、单元测试以及如何使用 Python 调用 Selenium WebDriver 接口的具体实例的材料相对零散。直到《*Learning Selenium Testing Tools with Python*》中文版的出版，使得我们有机会较为全面，并且系统性地学习用单一脚本语言开发 Web 自动化测试的具体实践，作者独特的创作逻辑，使得本书前后实例相互对照，并且首尾呼应。既诠释原理，又能使读者进入实战，还有“干货”满满的“提醒与备注”，是一本不可多得的自动化测试指导书。这也是我们译著这本书最重要的原因了。

本书的作者 Unmesh Gundecha 有着极为丰富构建自动化测试解决方案的经验。主导开发过大量商业或开源的自动化测试工具。曾供职微软。在 2012 年编著过《*Selenium Testing Tools Cookbook*》一书，颇为畅销；在 2015 年下半年又更新发布了第二版。作者文笔犀利，逻辑之间环环相扣，语言诙谐，妙笔生花。

多年的技术文章翻译经验，使我们清晰地认识到，倘若停留在专业翻译层面，想必本书的可读性，以及作者的诸多表述，都难以顺利地传递给中文读者。所以我与熊志男（本书合

译者), 多次调整翻译策略。由传统的分章节翻译, 到“按作者编著脉络”组织分工。由专业词汇翻译, 到统一关键词口径, 甚至到整句、整段打乱重组。诸如此类的一些做法, 都是为了保证我们的译著质量更加符合测试同行的阅读习惯, 便于学习与加深印象。

好在翻译过程中, 与熊志男相互鼓励, 包括审核团队不厌其烦的讨论、PK、一起揣摩作者意图。大家的专业、包容、豁达、自信, 让这本书的中文译著得以完成。特别感谢参与翻译工作的张欣欣、谢满彬、谢柳娜。感谢测试窝网译文团队的多次审校。

翻译别人的图书, 好似在反刍, 再精彩也是在讲别人的故事。期待有一天, 有机会能够讲讲我们自己的故事给广大同行。由于译者的水平有限, 难免会有偏差疏漏。若有欠妥之处, 欢迎指正, 编辑联系邮箱 [zhangtao@ptpress.com.cn](mailto:zhangtao@ptpress.com.cn)。

金 鑫



# 业界评价

在互联网行业迅速发展的今天，编写自动化脚本的技能，已经逐渐成为 Web 测试人员的标配。

Python 作为备受测试人员青睐的语言之一，非常适合处理日常工作中的数据和文本问题。

Selenium 更是 UI 自动化测试的利器，但要迅速掌握并熟练运用到项目中，绝非易事。

本书围绕 Selenium 的使用展开，编排有序，通俗易懂，对于没有 UI 自动化测试经验的读者，将起到事半功倍的效果。

——Ping++ 质量负责人 吴子腾

Unmesh Gundecha 编著的《*Selenium Testing Tools Cookbook*》，俗称“Selenium 菜谱”，是我一直推荐给身边 WebDriver 初学者的书籍，只是很遗憾一直未被翻译成中文版本出版。

该书作为“菜谱”的 Python 姊妹篇，秉承了“菜谱”的内容详实，案例丰富，行文流畅等特点，是一本 WebDriver 入门的绝佳教材。

——陈冬严，浙江大学硕士，具有 10 年软件测试和团队管理的工作经验，先后服务于领先的 ITSM、PLM 软件研发企业，现于某金融行业核心机构 IT 规划部门担任项目管理工作。业余时间喜欢园艺。《精通自动化测试框架设计》一书的作者。

# 关于作者

**Unmesh Gundecha** 拥有计算机软件硕士学位，在软件开发与测试领域有着 12 年的工作经验。无论是在应对业界标准，还是定制需求下，他都有着丰富的构建自动化测试解决方案的经验。与此同时，他还主导开发了大量商业或开源的自动化测试工具。

他曾供职于微软公司，从事开发有关的工作。目前在印度的一家跨国企业从事测试架构师工作，在 Ruby、Java、iOS、Android 和 PHP 的项目中有着极丰富的开发与测试经验。

# 作者语

另外，本书能顺利编写完成，离不开很多技术同行的帮助与审阅，感谢他们花费了大量的时间为本书提供了非常有价值的反馈。

感谢各位专家、同事和朋友，特别是 Yuri Weinstein 给予我很多帮助与鼓励。

# 关于审稿人

**Adil Imroz** 是一位 Python 的狂热爱好者，长期专注在测试开发与移动端自动化领域。崇尚开源与敏捷模式。闲暇时，爱好单车、读书、睡觉。他觉得这些都可以为他开拓眼界。

**Dr. Philip Polstra**（熟悉他的人都称呼他 Dr. Phil），国际知名黑客。他的作品曾在许多国际的专题会议（包括 DEFCON、Black Hat、44CON、Maker Faire 等）上提及，发表过大量的论文，是这一领域公认的专家。

**Dr. Polstra** 作为布鲁斯伯格大学的副教授，除了日常教学，还对外提供一些关于渗透测试的培训、咨询工作。

**Walt Stoneburner**，软件架构师，在商业应用开发与咨询领域有着 25 年以上的经验，另外在软件质量保证、配置管理与安全领域也有着长期的研究。

无论是在程序设计、协作应用、大数据、知识管理、数据可视化，还是在 ASCII 方面，他都有着很深的造诣。甚至在软件评测、消费电子产品测评、绘画、经营摄影工作室、创作幽默剧、游戏开发、无线电等领域都能找到他的身影，他还自称“极客”。

**Yuri Weinstein**，生活在旧金山，有超过 20 年的时间就职于硅谷顶尖的技术公司，专注测试领域，尤其是在自动化测试方向。目前在红帽公司负责 Ceph 开源存储项目的产品质量。

# 前言

Selenium 是一个主要用于 Web 应用程序自动化测试的工具集合，在行业内已经得到广泛的应用。然而其作用不局限于测试领域，还可以用于屏幕抓取与浏览器行为模拟等操作。它支持主流的浏览器，包括 Firefox、IE、Chrome、Safari 以及 Opera 等。

Selenium 包括一系列的组件。

- Selenium IDE: 是嵌入到 Firefox 浏览器的插件, 用于在 Firefox 上录制与回放 Selenium 脚本。图形化的界面可以形象地记录下用户在浏览器中的操作, 非常方便使用者了解与学习。目前它只能在 Firefox 下使用, 好在它能将录制好的脚本转换成各种 Selenium WebDriver 支持的程序语言, 进而扩展到更广泛的浏览器类型。
- Selenium WebDriver: 其实质上就是可以支持多种编程语言, 并且有用于操作浏览器的一套 API。支持多类型浏览器、跨操作系统平台 (包括 Linux、Windows 以及 Mac OS X), 是真正意义上的跨浏览器测试工具。WebDriver 为诸如 Java、C#、Python、Ruby、PHP、JavaScript 等语言分别提供了完备的、用于实现 Web 自动化测试的第三方库。
- Selenium Standalone Server: 包括被大家广泛了解的 Selenium Grid、远程控制、分布式部署等, 均可实现 Selenium 脚本的高效执行与拓展。我们利用 Grid 使得自动化测试可以并行运行, 甚至是在跨平台、异构的环境中运行, 包括目前主流的移动端环境, 如 Android、iOS。

正如书名所述, 这是一本介绍如何用 Python 语言调用 Selenium WebDriver 接口, 进而实

现对 Web 应用自动化测试的指导书。本书描述了从 Selenium 安装配置到基本使用，再到创建、调试、运行自动化脚本等进阶的操作。当然在开始之前，你可能需要先具备一定的 Python 语言基础。

## 内容介绍

**第 1 章基于 Python 的 Selenium WebDriver 入门** 从安装 Python、Selenium WebDriver 开始，到我们如何选择适合的 Python 编辑器，以及我们小试牛刀的第一个自动化测试脚本，并且成功地将这一脚本运行在不同浏览器上。

**第 2 章使用 unittest 编写单元测试** 本章带领我们结合 unittest 实现单元测试。通过转换后的脚本，有助于我们更好地完善单元测试用例。借助 unittest 实现测试用例集的整体运行，并将 HTML 格式的测试结果及时推送给项目的相关人员。

**第 3 章元素定位** 本章告诉你如何通过浏览器自带的开发者模式去定位页面中各类型元素。Selenium 通过获取这些元素的定位，进而实现模拟浏览器操作与参数捕获。这一章你将学会各种定位元素的方法，包括 XPath 和 CSS 以及对应的示例。

**第 4 章 Selenium Python API 介绍** 学习如何通过 WebDriver 与包括页面元素、JavaScript 提示框、框架 (frames)、窗口在内的各类对象进行交互，以及怎样进行浏览器回放、元素传值、鼠标点击、下拉菜单选择、多窗口切换等具体操作。

**第 5 章元素等待机制** 介绍多种设置等待方法，用于提高 Selenium 自动化测试脚本的稳定运行。带你理解显式等待或隐式等待的方法是如何应用于我们的测试脚本。

**第 6 章跨浏览器测试** 我们将深入学习如何在远程机器或 Selenium Grid 上通过 Remote WebDriver 实现测试脚本跨各类型浏览器的测试。Selenium Grid 可使得我们在多浏览器与多操作系统的排列组合中兼容测试，甚至支持像 PhantomJS 这样的无 UI 界面的浏览器。本章的最后，我们还将了解 Sauce Labs 和 BrowserStack 等第三方外部测试服务 (云测试)。

**第 7 章移动端测试** 我们使用 Selenium WebDriver、Appium 实现在包括 iOS 端、Android 端以及 Android 模拟器在内的移动设备上的自动化测试。另外，本章还有 App 测试的具体示例。

**第 8 章 Page Object 与数据驱动测试** 介绍这两种重要的设计模式，引导我们搭建更持续、更高效的测试框架。其中，Page Object 设计模式可帮助我们实现对界面细节的封装，并将一组用户行为构建在单个类中，提升自动化测试脚本的易读性和可复用性，从而达到更适应 UI



的频繁变化的目的。另外，我们还将学习用 `unittest` 实现数据驱动测试。

第 9 章 Selenium WebDriver 的高级特性 包括复杂的鼠标与键盘操作、cookies 操作、窗口截屏，甚至录制整个测试过程。

第 10 章第三方工具与框架集成 通过 Selenium 与持续集成工具的搭配，我们可以轻松地搭建自动化验收测试框架。本章中展示了“通过 Selenium 创建自动化验收测试用例，然后细化基于 UI 的自动化测试脚本，最后配置持续集成工具 Jenkins，最终实现了对被测程序每日构建、每日自动化验收测试的联动效果”的典型案列。

通过对本书的学习，你将能够用 Python 语言通过调用 Selenium WebDriver 接口，搭建属于你自己的 Web 应用自动化测试框架。

## 阅读前的准备工作

在阅读本书之前，你需要掌握 Python 语言基本语法以及 Web 前端的相关知识(如 HTML、JavaScript、CSS 和 XML)。如果你能编写一些简单的包括循环、条件判断、定义类等语法的 Python 脚本，你就能轻松地理解本书中的示例代码。每行示例代码我们都花了很大精力去注释说明，就是希望你能达到最佳的学习效果。还有一些前期准备的软件、工具以及环境配置都在第 1 章有明确的说明，你需要在你的机器上准备好访问终端、Python 解释器以及浏览器。

## 适合哪些人阅读

如果你从事 QA 或软件测试、软件开发、Web 应用开发等相关工作，希望用 Python 语言调用 Selenium WebDriver，以实现 Web 应用的自动化测试，那么这本书一定是你较好的选择！在学习 Selenium 理论之前，我们建议你掌握 Python 语言的基本语法。通过整本书的通篇学习，你将全面地理解 Selenium WebDriver 的相关知识，并且能有效地帮助你实现自动化测试。

## 约定

本书中，你可能会发现不同类型的信息，呈现出的文本风格不尽相同。我们在这里将罗列不同类型的文本风格以及对应的含义，方便你阅读。

代码文本的样式如下。

```
# create a new Firefox session
driver = webdriver.Firefox()
driver.implicitly_wait(30)
driver.maximize_window()
```

当我们想格外强调代码中的一部分时，相应的代码字体会被加粗，示例如下。

```
# run the suite
xmlrunner.XMLTestRunner(verbosity=2,output='test-reports').
run(smoke_tests)
```

命令行的输入\输出的样式如下。

```
pip install -U selenium
```

新的措辞与关键语句会被显示为粗体。关键语句就是指出现在系统界面、菜单项或对话框等位置的关键操作，例如，“在 **Tools** 下拉菜单中选择 **Internet Options**”。



警告或重要的提示，会出现在这样的括号中。



提醒或小窍门，会出现在这样的括号中。

## 读者反馈

我们十分乐见读者的反馈，让我们了解你对本书的想法——包括好与不好的评价。因为你的反馈将使我们以后可以更好地为读者提供有价值的内容。

可以通过我们的邮箱地址 [contact@epubit.com.cn](mailto:contact@epubit.com.cn) 将你的反馈信息告诉我们，邮件的主题需要注明书籍的全名。

当然，如果你也有专注的主题，并且有兴趣编辑或撰写图书，欢迎你联系我们，邮箱为 [zhangtao@ptpress.com.cn](mailto:zhangtao@ptpress.com.cn)。

# 目录

第 1 章 基于 Python 的 Selenium	
WebDriver 入门	1
1.1 环境准备	2
1.1.1 安装 Python	3
1.1.2 安装 Selenium 包	3
1.1.3 浏览 Selenium WebDriver Python 文档	3
1.1.4 选择一个 IDE	4
1.1.5 PyCharm 设置	8
1.2 第一个 Selenium Python 脚本	11
1.3 支持跨浏览器	16
1.3.1 设置 IE 浏览器	16
1.3.2 设置 Google Chrome 浏览器	19
1.4 章节回顾	21
第 2 章 使用 unittest 编写单元测试	22
2.1 unittest 单元测试框架	23
2.1.1 TestCase 类	25
2.1.2 类级别的 setUp()方法和 tearDown()方法	30
2.1.3 断言	32
2.1.4 测试套件	33
2.2 生成 HTML 格式的测试报告	36
2.3 章节回顾	38
第 3 章 元素定位	39
3.1 借助浏览器开发者模式定位	42
3.1.1 用火狐浏览器 Firebug 插件 检查页面元素	42
3.1.2 用谷歌 Chrome 浏览器 检查页面元素	43
3.1.3 用 IE 浏览器检查页面 元素	44
3.2 元素定位	45
3.2.1 ID 定位	46
3.2.2 name 定位	47
3.2.3 class 定位	47
3.2.4 tag 定位	48
3.2.5 XPath 定位	50
3.2.6 CSS 选择器定位	51
3.2.7 Link 定位	53