

Broadview[®]
www.broadview.com.cn

这不仅是一本接口自动化测试书
同时还是一本Django Web开发书

Web接口开发与

自动化测试

基于Python语言

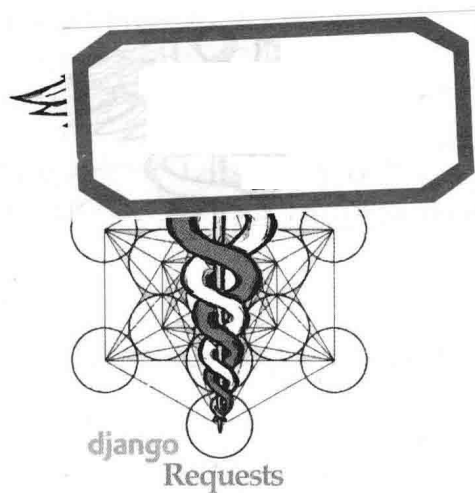
虫师◎编著



django
Requests

 中国工信出版集团

 电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>



Web接口开发与 自动化测试 基于Python语言

虫师◎编著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书以接口测试为主线，以 Web 开发为切入点，全面介绍了 Web 接口开发与自动化测试过程中使用的相关技术。全书分为 15 章，第 1 章介绍了 Python 的基础知识，解答初学 Python 的同学都会遇到的一些问题；第 2 章到第 5 章以开发发布会签到系统为需求，介绍了 Django Web 开发技术；第 6 章介绍了 unittest 单元测试框架，以及在 Django 中如何编写单元测试；第 7 章到第 10 章主要围绕接口的相关概念，Web 接口开发，接口测试工具和接口自动化测试框架的开发；第 11 介绍了安全接口的开发与测试；第 12、13 章介绍了 Web Services 技术；第 14、15 章介绍了 Django Web 项目的部署和接口性能测试。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

图书在版编目 (CIP) 数据

Web 接口开发与自动化测试：基于 Python 语言 / 虫师编著. —北京：电子工业出版社，2017.4
ISBN 978-7-121-31099-7

I. ①W… II. ①虫… III. ①网页制作工具—程序设计②软件工具—程序设计 IV. ①TP393.092.2
②TP311.561

中国版本图书馆 CIP 数据核字(2017)第 050485 号

责任编辑：安 娜

印 刷：北京中新伟业印刷有限公司

装 订：北京中新伟业印刷有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编：100036

开 本：787×980 1/16 印张：18 字数：366 千字

版 次：2017 年 4 月第 1 版

印 次：2017 年 4 月第 1 次印刷

印 数：3000 册 定价：59.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010) 88254888, 88258888。

质量投诉请发邮件至 zlls@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：010-51260888-819, faq@phei.com.cn。

前 言

本书的原型是我整理的一份 Django 学习文档。在从事软件测试工作的这六七年里，我一直有整理学习资料的习惯，这种学习理解再输出的方式对我非常受用，博客和文档是我主要的输出形式，这些输出同时也帮助到许多软件测试人员。

说回到接口测试的话题上来，根据分层自动化测试的思想，上层为 UI 层。关于 UI 层的自动化测试我们已经很熟悉了，许多技术和工具都是围绕这一层来设计的，我们要想做自动化，首先想到和尝试去做的也是这一层的自动化实现。底层为单元测试，对于单元级别的自动化测试来说，虽然大多数测试人员并没有真正做过，但我们知道，它一般使用单元测试框架，通过一段代码去测试另一段代码；而接口测试刚好处于中间层，不太好理解，也不太好解释，因为在开发的项目中只有程序目录/文件、类、方法、函数这些，并没有一个叫作“接口”的东西。但是，它却又无处不在，是一个出现频率极高的词，时刻挂在开发人员的嘴边。

什么是接口？如何对接口进行测试呢？我曾经也有很长一段时间并不太理解什么是接口测试。为什么没有一本讲接口测试的书呢？性能测试和 UI 自动化测试的书籍每年都会出版好几本，与之相比，几乎找不到一本专门介绍接口测试的书。现在想来可能是因为它太简单了吧！简单到没什么可讲的，甚至接口测试比功能测试还要简单得多。但是，真的做好接口测试又很难，或者说难点并不在于接口测试本身，而是需要有读懂接口代码处理逻辑的能力，这就要求你必须具备一定的开发基础，因而对许多测试人员来说，已经形成了门槛。

要写一本关于接口测试的书，一种讨巧的做法就是把目前主流的接口测试工具都介绍一遍，这很符合主题。然而，我并不认为把这些工具都学好就可以做好接口测试。举一个简单的例子，我的接口使用了加密，例如，用时间戳+密钥生成 AES 加密字符串，再将字符串生成 base64 字符串作为接口参数传输，这其实是一种常见的加密策略，而我所了解的

大多数接口测试工具都无法做到对这种加密接口的测试。原因很简单，加密算法和加密策略多种多样，而工具却很难模拟这种多样性的加密策略。但是，站在开发的角度的看，接口测试就很简单了，开发怎么调用，测试就怎么调用呗！只不过测试的目的是验证在传各种参数的情况下，接口是否可以正确地处理并返回结果。

所以，我选择另一种需要很高学习成本的方法来讲解接口测试，从 Web 接口开发讲起，理解了接口是如何开发的，再做接口测试自然就变得非常简单了。你会看到本书前几章都是在讲 Web 开发以及 Web 接口开发，你可能会怀疑自己是不是买错书了，明明是要学习接口测试的，怎么介绍的都是开发的技术。其实，如果你只是想学习 Web 开发的话，那么本书也是一本不错的入门教程。

本书共分为 15 个章节，涵盖了不少话题，这也是我在写作时对自己的要求，不要讲解太基础的东西，不要讲太多无聊的概念，要有很强的可操作性。第 1 章是 Python 基础，第 2~5 章是 Web 开发，第 6 章是单元测试，第 7~10 章主要讲接口的概念、开发与测试，第 11 章介绍安全接口的开发与测试，第 12~13 章讲 Web Service 技术，第 14~15 章介绍项目的部署与性能测试。所以，这并不是一本单纯面向测试人员的书，同样适合开发的小伙伴阅读。

到了感谢部分，首先，感谢身边的同事，一年多的接口自动化测试实践过程中，我收获了很多，感谢测试经理唐亮对我们在尝试新技术时的支持，感谢开发组的蓝仕坤、陈晓发在技术上给了我很大的帮助。其次，感谢接口自动化测试群里的小伙伴，他们给本书提了很多建议，包括本书的名字，也是在群里投票的结果。再次，感谢我的妻子，她默默地容忍着我每天晚睡的坏习惯，感谢她一直以来的包容与理解。最后，感谢编辑安娜，没有她这本书也不会出版，我们合作一直很愉快。

由于作者水平有限，希望你带着怀疑的精神阅读本书，如果发现错误，欢迎批评指正。

2017 年 1 月 13 号凌晨

虫师

目 录

第 1 章 Python 学习必知.....	1
1.1 Python 2.x 与 Python 3.x 选择.....	1
1.2 Python 的安装.....	2
1.2.1 在 Windows 下安装 Python.....	2
1.2.2 安装 Python 2 和 Python 3 两个版本.....	3
1.2.3 “python”不是内部或外部命令.....	5
1.3 扩展库的安装.....	6
1.3.1 pip 安装扩展库.....	6
1.3.2 tar.gz 文件安装.....	9
1.3.3 .whl 文件安装.....	9
1.3.4 GitHub 克隆项目安装.....	9
1.4 开发工具选择.....	11
1.4.1 Sublime Text3.....	11
1.4.2 Atom.....	13
1.4.3 PyCharm.....	14
1.5 程序报错不要慌.....	15
1.5.1 缩进错误.....	15
1.5.2 引包错误.....	16
1.5.3 编码错误.....	16
1.5.4 学会分析错误.....	17
第 2 章 Django 入门.....	19
2.1 Django 开发环境.....	19
2.1.1 在 Windows 下安装 Django.....	20

2.1.2	在 Ubuntu 下安装 Django.....	20
2.2	开始第一个 demo	21
2.2.1	创建项目与应用	22
2.2.2	运行项目	25
2.2.3	Hello Django!	27
2.2.4	使用模板	29
2.3	Django 工作流.....	30
2.3.1	URL 组成	31
2.3.2	URLconf.....	32
2.3.3	views 视图.....	33
2.3.4	templates 模板	34
2.4	MTV 开发模式	34
第 3 章	Django 视图	36
3.1	来写个登录功能	36
3.1.1	GET 与 POST 请求.....	37
3.1.2	处理登录请求	40
3.1.3	登录成功页	42
3.2	Cookie 和 Session.....	44
3.2.1	Cookie 的使用.....	44
3.2.2	Session 的使用	46
3.3	Django 认证系统.....	49
3.3.1	登录 Admin 后台	49
3.3.2	引用 Django 认证登录	50
3.3.3	关上窗户	51
第 4 章	Django 模型	53
4.1	设计系统表	53
4.2	admin 后台管理	56
4.3	基本数据访问	59
4.3.1	插入数据	60
4.3.2	查询数据	61

4.3.3	删除数据	63
4.3.4	更新数据	63
4.4	SQLite 管理工具	64
4.4.1	SQLite Manager	64
4.4.2	SQLiteStudio	65
4.5	配置 MySQL	65
4.5.1	安装 MySQL	65
4.5.2	MySQL 基本操作	67
4.5.3	安装 PyMySQL	68
4.5.4	在 Django 中配置 MySQL	69
4.5.5	MySQL 管理工具	72
第 5 章	Django 模板	73
5.1	Django-bootstrap3	73
5.2	发布会管理	74
5.2.1	发布会列表	74
5.2.2	搜索功能	78
5.3	嘉宾管理	79
5.3.1	嘉宾列表	80
5.3.2	分页器	83
5.4	签到功能	87
5.4.1	添加签到链接	87
5.4.2	签到页面	88
5.4.3	签到动作	91
5.5	退出系统	93
第 6 章	Django 测试	95
6.1	unittest 单元测试框架	95
6.1.1	单元测试框架	95
6.1.2	编写单元测试用例	96
6.2	Django 测试	100
6.2.1	一个简单的例子	100

6.2.2	运行测试用例	102
6.3	客户端测试	104
6.3.1	测试首页	104
6.3.2	测试登录动作	105
6.3.3	测试发布会管理	107
6.3.4	测试嘉宾管理	108
6.3.5	测试用户签到	109
第 7 章	接口相关概念	111
7.1	分层的自动化测试	111
7.2	单元测试与模块测试	112
7.3	接口测试	114
7.3.1	接口的分类	115
7.3.2	接口测试的意义	116
7.4	编程语言中的 Interface	117
7.4.1	Java 中的 Interface	117
7.4.2	Python 中的 Zope.interface	119
第 8 章	开发 Web 接口	121
8.1	为何要开发 Web 接口	121
8.2	什么是 Web 接口	124
8.2.1	HTTP	125
8.2.2	JSON 格式	128
8.3	开发系统 Web 接口	129
8.3.1	配置接口路径	129
8.3.2	添加发布会接口	130
8.3.3	查询发布会接口	132
8.3.4	添加嘉宾接口	133
8.3.5	查询嘉宾接口	135
8.3.6	发布会签到接口	136
8.4	编写 Web 接口文档	138

第 9 章 接口测试工具介绍	143
9.1 Postman 测试工具	143
9.2 JMeter 测试工具	146
9.2.1 安装 JMeter	146
9.2.2 创建测试任务	147
9.2.3 添加接口测试	151
9.2.4 添加断言	153
9.3 Robot Framework 测试框架	154
9.3.1 环境搭建	155
9.3.2 基本概念与用法	157
9.3.3 接口测试	160
第 10 章 接口自动化测试框架	165
10.1 接口测试工具的不足	165
10.2 Requests 库	166
10.2.1 安装	167
10.2.2 接口测试	167
10.2.3 集成 unittest	168
10.3 接口测试框架开发	169
10.3.1 框架处理流程	169
10.3.2 框架结构介绍	170
10.3.3 修改数据库配置	171
10.3.4 数据库操作封装	172
10.3.5 编写接口测试用例	176
10.3.6 集成测试报告	178
第 11 章 接口的安全机制	181
11.1 用户认证	181
11.1.1 开发带 Auth 接口	182
11.1.2 接口文档	184
11.1.3 接口测试用例	185

11.2	数字签名	187
11.2.1	开发接口	188
11.2.2	接口文档	191
11.2.3	接口用例	192
11.3	接口加密	194
11.3.1	PyCrypto 库	194
11.3.2	AES 加密接口开发	196
11.3.3	编写接口文档	201
11.3.4	补充接口测试用例	202
第 12 章	Web Services	205
12.1	Web Services 相关概念	205
12.2	Web Services 的开发与调用	214
12.2.1	suds-jurko 调用接口	214
12.2.2	spyne 开发接口	219
12.3	JMeter 测试 SOAP 接口	221
第 13 章	REST	224
13.1	RPC 与 REST	224
13.2	Django REST Framework	227
13.2.1	创建简单的 API	227
13.2.2	添加接口数据	231
13.2.3	测试接口	232
13.3	集成发布会系统 API	234
13.3.1	添加发布会 API	234
13.3.2	测试接口	237
13.4	soapUI 测试工具	238
13.4.1	创建 SOAP 测试项目	239
13.4.2	创建 REST 测试项目	241

第 14 章 Django 项目部署	244
14.1 uWSGI	244
14.1.1 uWSGI 介绍	244
14.1.2 安装 uWSGI	245
14.1.3 uWSGI 运行 Django	246
14.2 Nginx	247
14.2.1 安装 Nginx	247
14.2.2 Nginx+uWSGI+Django	248
14.2.3 处理静态资源	251
14.3 创建 404 页面	253
第 15 章 接口性能测试	256
15.1 Locust 性能测试工具	256
15.1.1 安装 Locust	257
15.1.2 性能测试案例	259
15.2 发布会系统性能测试	262
15.2.1 性能测试准备	263
15.2.2 编写性能测试脚本	266
15.2.3 执行性能测试	267
15.3 接口性能测试	270
15.3.1 编写接口性能测试脚本	270
15.3.2 执行接口性能测试	271
15.3.3 多线程测试接口性能	274

第 1 章

Python 学习必知

本书将以 Python 编程语言为基础来介绍开发与测试技术，所以，在阅读本书之前要求读者具备一定的 Python 语言编程能力。以我个人学习 Python 的经历，以及帮助别人解答 Python 问题的经验来看，对于初学 Python 的人来说，遇到的大多数问题并不是 Python 的语法，如果读者稍有编程语言基础，那么很容易就能学会 Python 的语法，而且对于这方面的学习，可以轻松地找到大量的文章、书籍和视频教程等。然而，他们问的更多的问题是 Python 版本的选择、环境的设置、第三方扩展库的安装、IDE 的选择，以及遇到程序报错之后怎么解决等。那么，本章将试着帮你清除这些障碍，使你后续的 Python 学习过程变得更加顺利。

1.1 Python 2.x 与 Python 3.x 选择

对于想要学习 Python 的同学来说、首先要面对的就是版本选择的问题。到底是选择学习 Python 2.x 还是 Python 3.x？这主要由 Python 语言发展的历史遗留问题所导致。

Python 语言早在 1989 由 Guido van Rossum 开发，第一个公开发行人版发行于 1991 年。因为早期的 Python 版本在基础方面设计存在着一些不足之处。因此在 2008 年的时候，Guido van Rossum 又重新发布了 Python 3.0，Python 3 在设计的时候很好地解决了这些遗留问题，然而 Python 3 带来的最大的问题就是不完全向后兼容，当时向后兼容的版本是 Python 2.6。然而经过多年的发展，Python (2.0 版本) 已经成为了一门应用非常广泛的编程语言，大量的项目在 Python 语言上运行，围绕着 Python 语言有着极其丰富的类库，无法一下子就让所有项目和类库都转到 Python 3.0 上面。于是，两个版本就进入了长期并行开发和维护的阶段。

正是由于官方对 Python 2.x 的纵容态度，致使到目前为止，Python 2 的使用者依然过半。

从近两年来看，官方的态度有所改变，Python 2.x 的开发逐渐进入消极状态，版本更新速度明显要比 Python 3.x 要慢得多，而且不再加入新的特性，以维护为主。Python 语言作者 Guido van Rossum 宣布 Python 2.7 支持时间延长到 2020 年。Python 2.7 是 2.x 系列的最后一个版本。这将有利于 Python 3 的发展。

对于新手来说，建议直接学习 Python 3.x，因为 Python 3.x 代表了 Python 发展的未来。目前主流的库基本都已支持 Python 3.x，不支持的库也在积极地向 Python 3.x 迁移。在本书中除非特别声明，否则默认情况下所有代码都将在 Python 3.x 下运行。

注：Python 3.x 和 Python 2.x，x 表示小版本号。当前 Python 最新的两个稳定版本分别为 Python 3.5.2 和 Python 2.7.12。如果没有特别说明，本书将以 Python 3 来指代 Python 3.x，用 Python 2 指代 Python 2.x。

1.2 Python 的安装

Python 的安装相当简单，但仍有一些细节需要注意。

1.2.1 在 Windows 下安装 Python

Python 下载地址：<https://www.python.org/downloads/>。

当前最新版本为 Python 3.5.2。读者可根据自己的系统平台选择相应的版本进行下载。对于 Windows 用户来说，如果是 32 位系统则选择 x86 版本；如果是 64 位系统，则选择 x86-64 版本。建议选择“executable installer”版本下载，下载完成后会得到一个以.exe 为后缀的文件，双击进行安装，如图 1.1 所示。



图 1.1 Python 安装界面

安装过程中记得勾选“Add Python 3.5 to PATH”选项。安装完成后将会在开始菜单中生成 Python 3.5 的目录，如图 1.2 所示。

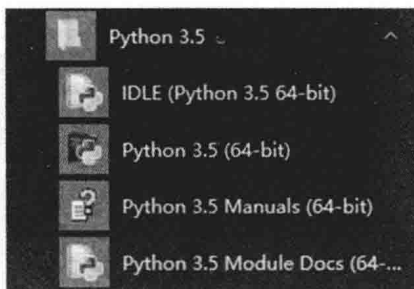


图 1.2 Python 3.5 的目录

在 Windows 系统中，安装好的 Python 提供了四个选项：

- ◎ IDLE (Python 3.5 64-bit)：Python 自带的 IDE，推荐新手使用它来编写 Python 程序。
- ◎ Python 3.5 (64-bit)：在 Windows 命令提示符下进入 Python Shell 模式。
- ◎ Python 3.5 Manuals (64-bit)：Python 自带的官方文档。
- ◎ Python 3.5 Module Docs (64-bit)：Python 的模块文档。它自动启动一个服务，以 Web 形式显示 Python 模块的文档。

1.2.2 安装 Python 2 和 Python 3 两个版本

虽然 Python 3 正在逐渐取代 Python 2，但是从目前来看，Python 2 的使用者仍然过半，除 Python 2 的坚定拥护者外，最主要的原因是仍有少部分的第三方库还不支持 Python 3，但这种情况在不断改善中。所以，有时为了使用某个库而不得不在两个版本之间进行切换。这时就需要在系统中同时安装两个 Python 版本了。

当然，Python 早就考虑到了可能会有这样的需求，所以，它允许你在一个操作系统中同时安装两个版本。并且，主流 Linux（例如 Ubuntu）系统已经默认为你安装了两个版本的 Python。如果是 Windows 系统，那么你需要手动来安装两个版本的 Python。不过，在使用两个版本的时候，需要做好区分。

例如，我本机先安装的 Python 2.7，如图 1.3 所示。



图 1.3 Python 2.7 目录

在 Python 2.7 的根目录下，Python 的可执行文件名为“python.exe”。当要运行 Python 2.7 版本时，只需在 Windows 命令提示符下输入“python”命令即可，如图 1.4 所示。

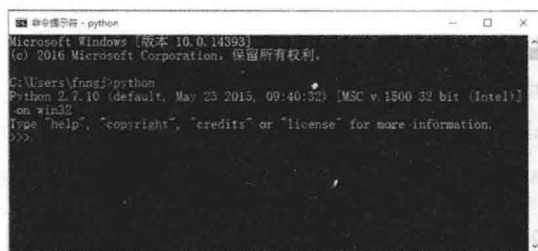


图 1.4 进入 Python 2.7 Shell 模式

然后，又安装了 Python 3.5。再来看看 Python 3.5 的目录，如图 1.5 所示。



图 1.5 Python 3.5 目录

除了生成一个“python.exe”文件外，还多了一个“python3.exe”文件。此时，如果想运行 Python 3.5，就可以使用“python3”命令，如图 1.6 所示。

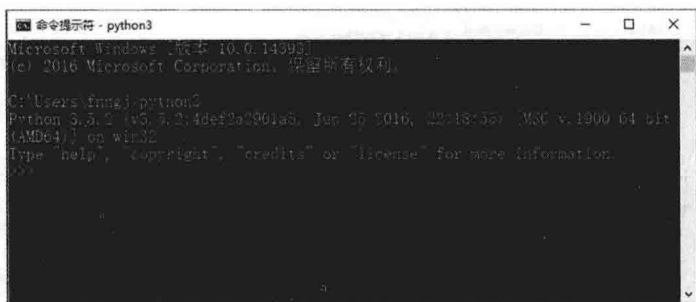


图 1.6 进入 Python 3.5 Shell 模式

需要说明的是，Python 可执行文件“python.exe”的名称是可以随意修改的。只要能有效区分两个 Python 版本即可。

注：如果没有特别说明，本书接下来的所有地方都将用“python 3”来指代 Python 3.5 的“python”命令。

1.2.3 “python”不是内部或外部命令

这个问题也是新手可能会遇到的，虽然在安装 Python 的时候，已经提醒你勾选“Add Python 3.5 to PATH”选项，但也许你并未留心这个提示，如图 1.7 所示。

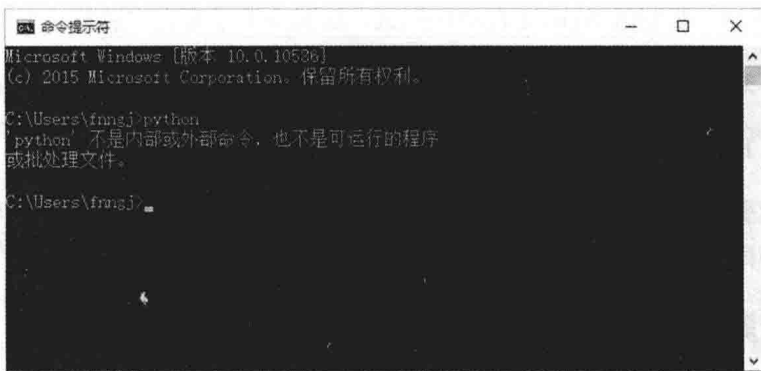


图 1.7 “python”不是内部或外部命令

此时先要确定 Python 安装到哪个目录。如图 1.3 和 1.5 所示，分别为我的 Python 2.7 和 Python 3.5 的安装目录，并且将它分别添加到系统环境变量 Path 下面，如图 1.8 所示。