PACKT PUBLISHING

异步图书 www.epubit.com.cn

使用Python 3.5学习编码实践以及高级概念

Python 高级编程

（第2版）

Expert Python Programming

Second Edition

[波兰] Michał Jaworski 著
[法] Tarek Ziadé
张亮 阿信 译

中国工信出版集团 人民邮电出版社 POSTS & TELECOM PRESS

Python高级编程

（第2版）

[波兰] Michał Jaworski 著
[法] Tarek Ziadé
张亮 阿信 译

人民邮电出版社
北京

图书在版编目（CIP）数据

Python高级编程 : 第2版 / (波) 贾沃斯基(Michal Jaworski), (法) 莱德 (Tarek Ziadé) 著 ; 张亮, 阿信译. -- 2版. -- 北京 : 人民邮电出版社, 2017.10
ISBN 978-7-115-46015-8

Ⅰ. ①P… Ⅱ. ①贾… ②莱… ③张… ④阿… Ⅲ. ①软件工具－程序设计 Ⅳ. ①TP311.561

中国版本图书馆CIP数据核字(2017)第151660号

版权声明

◆ 著　[波兰] Michał Jaworski　[法] Tarek Ziadé
译　张　亮　阿　信
责任编辑　胡俊英
执行编辑　武晓燕
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
北京鑫正大印刷有限公司印刷

◆ 开本：800×1000　1/16
印张：26.5
字数：525 千字　2017 年 10 月第 2 版
印数：3 001 – 6 000 册　2017 年 10 月北京第 1 次印刷

著作权合同登记号　图字：01-2016-7609 号

定价：89.00 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

Python 作为一种高级程序设计语言，凭借其简洁、易读及可扩展性日渐成为程序设计领域备受推崇的语言之一。

本书基于 Python 3.5 版本进行讲解，通过 13 章的内容，深度揭示了 Python 编程的高级技巧。本书从 Python 语言及其社区的现状开始介绍，对 Python 语法、命名规则、Python 包的编写、部署代码、扩展程序开发、管理代码、文档编写、测试开发、代码优化、并发编程、设计模式等重要话题进行了全面系统化的讲解。

本书适合想要进一步提高自身 Python 编程技能的读者阅读，也适合对 Python 编程感兴趣的读者参考学习。全书结合典型且实用的开发案例，可以帮助读者创建高性能的、可靠且可维护的 Python 应用。

作者简介

Michał Jaworski 有着 7 年 Python 编程的经验。他还是 graceful 的创建者，这是一个构建于 falcon 之上的 REST 框架。他曾在不同的公司担任过多种角色，从一名普通的全栈开发人员到软件架构师再到一家快节奏创业公司的工程副总裁。他目前是 Opera 软件公司 TV Store（电视应用商店）团队的首席后端工程师。他在设计高性能的分布式服务方面拥有丰富的经验。他还是一些流行的 Python 开源项目的活跃贡献者。

Tarek Ziadé是 Mozilla 的工程经理，与一个专门用 Python 为 Firefox 构建大规模 Web 应用的团队合作。他对 Python 打包做出过贡献，而且从早期 Zope 开始就使用过各种不同的 Python Web 框架。

Tarek 还创建了 Afpy——法国的 Python 用户组，并且用法语写过两本关于 Python 的书。他还在诸如 Solutions Linux、PyCon、OSCON 和 EuroPython 等国际活动中做过多次法语演讲和教学。

译者简介

张亮（hysic），毕业于北京大学物理学院，是一名爱好机器学习和数据分析的核安全工程师，主要负责本书前 6 章的翻译，并对本书进行了技术审读。

阿信，软件工程师，业余时间喜欢读书，也喜欢翻译。

审稿人简介

Facundo Batista 是 Python 编程语言方面的专家，拥有超过 15 年的 Python 编程经验。他是这门语言的核心开发者，也是 Python 软件基金会的成员。他还获得了 2009 年的社区服务奖，奖励他组织了阿根廷的 PyCon 及其 Python 社区，以及对标准库的贡献和在翻译 Python 文档方面所做的工作。

他还在阿根廷与其他国家（美国和欧洲）的主要 Python 会议上发表演讲。总之，他有丰富的分布式协同经验，10 多年来一直参与 FLOSS 开发并与全球人员合作。

他曾在 Telefónica Móviles 和 Ericsson 担任电信工程师，还曾在 Cyclelogic 担任 Python 专家（首席开发工程师），目前的职务是 Canonical 的高级软件开发工程师。

他还喜欢打网球，同时是两个可爱宝宝的父亲。

前言

Python 很棒！

从 20 世纪 80 年代末出现的最早版本到当前版本，Python 的发展一直遵循着相同的理念：提供一个同时具备可读性和生产力的多范式编程语言。

人们曾经将 Python 看作另一种脚本语言，认为它不适合构建大型系统。但多年以来，在一些先驱公司的努力下，Python 显然可以用于构建几乎任何类型的系统。

实际上，许多其他语言的开发者也醉心于 Python，并将它作为首选语言。

如果你购买了这本书，可能已经知道这些内容了，所以无需再向你证明这门语言的优点。

本书展现了作者多年构建各种 Python 应用的经验，从几个小时完成的小型系统脚本，到许多开发人员历经数年编写的大型应用。

本书描述了开发人员使用 Python 的最佳实践。

本书包含了一些主题，这些主题并不关注语言本身，而是更多地关注如何利用相关的工具和技术。

换句话说，本书描述了高级 Python 开发人员每天的工作方式。

本书内容

第 1 章介绍了 Python 语言及其社区的现状。本章展示了 Python 不断变化的方式及原因，还解释了为什么这些事实对任何想要自称 Python 专家的人来说是很重要的。本章还介绍了最流行和最公认的 Python 工作方式——常用的生产力工具和现已成为标准的约定。

第 2 章深入介绍迭代器、生成器、描述符等内容。本章还包括关于 Python 习语和 CPython 类型内部实现的有用注释，这些类型的计算复杂度是对这些习语的阐释。

第 3 章介绍了语法最佳实践，但重点放在类级别以上。本章包括 Python 中更高级的面向对象的概念和机制。学习这些知识是为了理解本章最后一节的内容，其中介绍的是 Python

元编程的各种方法。

第 4 章介绍了如何选择好的名称。它是对 PEP 8 中命名最佳实践的扩展，并且给出了一些如何设计良好 API 的提示。

第 5 章介绍如何创建 Python 包以及使用哪些工具，以便在官方的 Python 包索引或其他包仓库中正确地分发。对于 Python 包还补充了一些工具的简要回顾，这些工具可以让你用 Python 源代码创建独立可执行文件。

第 6 章主要针对 Python Web 开发人员和后端工程师，因为讲的是代码部署。本章解释了如何构建 Python 应用，使其可以轻松部署到远程服务器，还介绍了可以将这个过程自动化的工具。本章是第 5 章的延续，因此还介绍了如何使用包和私有包仓库来简化应用部署。

第 7 章解释了为什么为 Python 编写 C 扩展程序有时可能是一个好的解决方案。本章还展示了只要使用了正确的工具，它并不像想象中那么难。

第 8 章深入介绍了项目代码库的管理方式，还介绍了如何设置各种持续开发流程。

第 9 章包含文档相关的内容，提供了有关技术写作和 Python 项目文档化方式的建议。

第 10 章解释了测试驱动开发的基本原理，还介绍了可用于这种开发方法的工具。

第 11 章解释了何为优化，介绍了分析技术和优化策略指南。

第 12 章是对第 11 章的扩展，为 Python 程序中经常出现的性能问题提供了一些常用的解决方案。

第 13 章介绍了 Python 并发这一宏大的主题。本章解释了并发的概念、何时需要编写并发应用，以及 Python 程序员主要使用的并发方法。

第 14 章用一套有用的设计模式以及 Python 的代码示例对本书进行了总结。

阅读本书的前提

本书面向的是可以在任何操作系统上使用 Python 3 进行软件开发的人员。

这不是一本面向初学者的书，所以我假设你已经在开发环境中安装了 Python，或者知道如何安装 Python。不管怎样，本书考虑到以下事实：不是每个人都需要充分了解 Python 的最新功能或官方推荐的工具。因此，第 1 章概述了常见的实用程序（例如虚拟环境和 pip），这些实用程序现在已经成为 Python 专业开发人员的标准工具。

目标读者

本书面向的是想要进一步掌握 Python 的开发人员。开发人员主要指的是专业人士，即用 Python 编写软件的程序员。这是因为本书主要侧重于工具和实践，它们对于创建高性

能的、可靠且可维护的 Python 软件至关重要。

这并不意味着业余爱好者无法从本书中发现有趣的内容。对于任何对学习 Python 高级概念感兴趣的人来说，本书都是很棒的。任何具备 Python 基本技能的人都应该能够读懂本书的内容，虽然经验不足的程序员可能需要一些额外的努力。对于有点落后仍在继续使用 Python 2.7 或更老版本的人来说，本书也是对 Python 3.5 的全面介绍。

最后，从阅读本书中受益最多的人群应该是 Web 开发者和后端工程师。这是因为本书重点介绍了在他们的工作领域中特别重要的两个主题：可靠的代码部署与并发。

本书约定

本书用多种文本样式来区分不同种类的信息。下面是这些样式的示例及其含义解释。

文本中的代码、数据库表的名称、文件夹名称、文件名称、文件扩展名、路径名称、虚拟 URL、用户输入和 Twitter 句柄的格式如下所示：“利用 `str.encode(encoding, errors)`方法，用注册编解码器对字符串进行编码。”

代码块的格式如下所示：

```
[print("hello world")
print "goodbye python2"
```

如果我们想让你将注意力集中在代码块的特定区域，相关的几行或几项将会被设成粗体，如下所示：

```
cdef long long fibonacci_cc(unsigned int n) nogil:
    if n < 2:
        return n
    else:
        return fibonacci_cc(n - 1) + fibonacci_cc(n - 2)
```

命令行的输入或输出如下所示：

```
$ pip show pip
---
Metadata-Version: 2.0
Name: pip
Version: 7.1.2
Summary: The PyPA recommended tool for installing Python packages.
Home-page: https://pip.pypa.io/
Author: The pip developers
Author-email: python-virtualenv@groups.google.com
License: MIT
```

```
Location: /usr/lib/python2.7/site-packages
Requires:
```

新术语和**重要词语**将以粗体显示。你会在屏幕上看到的单词（例如在菜单或对话框中）将以下面这种文本形式出现：“单击 **Next** 按钮可跳转至下一屏”。

警告或重要提示。

提示和技巧。

读者反馈

我们十分欢迎读者的反馈意见。让我们了解你对本书的看法——喜欢哪些内容，不喜欢哪些内容。这些反馈对我们很重要，因为它有助于我们编写出对读者真正有帮助的书。

一般性的反馈请发送邮件至 feedback@packtpub.com，并在邮件主题中注明本书的标题。

如果你是某个领域的专家，并且有兴趣写一本书或者参与出版一本书，请参阅我们的作者指南。

客户支持

现在你已经成为这本 Packt 图书的拥有者，为了让你的购买物超所值，我们还为你提供了许多其他方面的服务。

下载示例代码

你可以用自己的账号在 Packt 的官方网站下载本书的示例代码文件。如果你是在其他地方购买的本书，可以访问 Packt 的官方网站并注册，文件会直接通过邮件发送给你。

下载代码文件的步骤如下所示。

- 用你的电子邮件地址和密码登录或注册我们的网站。
- 将鼠标指针悬停在顶部的 **SUPPORT** 选项卡上。
- 单击 **Code Downloads & Errata**。
- 在 **Search** 框中输入本书的名字。
- 选择你要下载代码文件的书籍。
- 从下拉菜单中选择本书的购买途径。

- 单击 **Code Download**。

你还可以在 Packt 网站的本书页面单击 **Code Files** 按钮来下载代码文件。在 **Search** 框输入本书的书名即可访问该页面。请注意，你需要登录 Packt 账号。

文件下载完成后，请确保用下列软件的最新版本对文件夹进行解压或提取。

- 在 Windows 上用 WinRAR 或 7-Zip。
- 在 Mac 上用 Zipeg、iZip 或 UnRarX。
- 在 Linux 上用 7-Zip 或 PeaZip。

本书的代码包也托管在 GitHub，网址为 https://github.com/PacktPublishing/Expert-Python-Programming_Second-Edition。在 GitHub 上还有大量图书和视频资源。快去看一下吧！

勘误

尽管我们已经竭尽全力确保本书内容的准确性，但错误在所难免。如果你发现了书中的错误，无论是正文错误还是代码错误，希望你能将其报告给我们，我们将不胜感激。这样不仅能够减少其他读者的困惑，还能帮助我们改进本书后续版本的质量。如果你需要提交勘误，请访问 http://www.packtpub.com/submit-errata，选择相应的书名，单击 **Errata Submission Form** 链接，然后输入你的勘误信息并提交。一旦通过验证，我们将接受你提交的勘误，同时勘误内容也将被上传到我们的网站，或者被添加到对应勘误区的现有勘误列表中。

想要查看之前提交的勘误，请访问 https://www.packtpub.com/books/content/support，并在搜索框中输入相应的书名。你想查看的信息将出现在 **Errata** 下面。

侵权行为

所有媒体在互联网上都一直饱受版权侵害的困扰。Packt 坚持对版权和授权进行全力保护。如果你在互联网上发现我社图书任何形式的盗版，请立即为我们提供网址或网站名称，以便我们采取进一步的措施。

请将疑似盗版材料的链接发送到 copyright@packtpub.com。

我们感谢你对作者的保护，这有助于我们继续为你提供更有价值的内容。

疑难解答

如果你对本书的某个方面抱有疑问，请通过 questions@packtpub.com 联系我们，我们会尽力为你解决。

目录