



Python 3

Object Oriented Programming

Python 3

面向对象编程

[加] Dusty Phillips 著

肖鹏 常贺 石琳 译



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

Python 3

Object Oriented Programming

Python 3

面向对象编程

[加] Dusty Phillips 著
肖鹏 常贺 石琳 译

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

Python 是一种面向对象的解释型语言，面向对象是其非常重要的特性。本书通过 Python 的数据结构、语法、设计模式，从简单到复杂，从初级到高级，一步步通过例子来展示了 Python 中面向对象的概念和原则。

本书不是 Python 的入门书籍，适合具有 Python 基础经验的开发人员阅读。如果你拥有其他面向对象语言的经验，你会更容易理解本书的内容。

Copyright ©Packt Publishing 2010. First published in the English language under the title 'Python 3 Object Oriented Programming'

本书简体中文版专有出版权由 Packt Publishing 授予电子工业出版社。未经许可，不得以任何方式复制或抄袭本书的任何部分。专有出版权受法律保护。

版权贸易合同登记号 图字：01-2015-0634

图书在版编目（CIP）数据

Python 3 面向对象编程 / (加) 菲利普斯 (Phillips,D.) 著；肖鹏，常贺，石琳译。—北京：电子工业出版社，2015.7

书名原文：Python 3 Object Oriented Programming

ISBN 978-7-121-26246-3

I. ①P… II. ①菲… ②肖… ③常… ④石… III. ①软件工具—程序设计 IV. ①TP311.56

中国版本图书馆 CIP 数据核字(2015)第 122883 号

策划编辑：张春雨 付 睿

责任编辑：付 睿

印 刷：北京中新伟业印刷有限公司

装 订：三河市华成印务有限公司

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：23.75 字数：478 千字

版 次：2015 年 7 月第 1 版

印 次：2015 年 7 月第 1 次印刷

定 价：79.00 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。

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

服务热线：(010) 88258888。

关于作者

Dusty Phillips 是一名加拿大自由软件开发人员、教师、武术家以及开源爱好者。他和 Arch Linux 社区以及其他开源社区有着紧密联系。他维护着 Arch Linux 的门户网站以及编译了流行的 Arch Linux 手册。Dusty 持有计算机科学人机交互方向的硕士学位。他目前在自己的计算机里安装了 6 种不同的 Python 解释器。

我要感谢我的编辑 Steven Wilding 和 Mayuri Kokate 对我适时的鼓励和反馈。感谢我的朋友及导师 Jason Chu 让我开始学习 Python，数年来耐心地回答我关于 Python、GIT 和生活中的各种问题。感谢我的父亲 C. C. Phillips，激励我在编辑他了不起的小说作品的同时进行写作。最后感谢每一个说迫不及待想买我的书的人；你们的热情对我是一股巨大的激励力量。

关于审校

Jason Chu 是 Oprius 软件公司联合创始人，CTO。他有超过 8 年的软件开发经验。Chu 从 2003 年版本 2.2 时就开始使用 Python。当他不开发个人或者专业软件的时候，会在空闲时间教空手道、下围棋，以及在他的家乡加拿大维多利亚 BC 玩。你会经常发现他在 Christie 马车房子里喝 Back Hand of God Stout 这种酒。

Michael Driscoll 用 Python 编程已近 4 年，从 20 世纪 90 年代末开始涉足其他语言。他从大学毕业获得科学学士学位，主修信息系统管理。Michael 为了兴趣和利益编程。他的爱好包含圣经教义，写关于 Python 的博客 <http://www.blog.pythonlibrary.org/>，以及学习摄影。Michael 目前在当地政府工作，一个能让他尽可能使用 Python 编程的地方。这是他作为技术审校的第一本书。

我要感谢我的妈妈，没有她我不可能成长成现在这样热爱学习的我。我还想感谢 Scott Williams 迫使我学习 Python，没有他，我甚至不知道这个语言的存在。最重要的是，我要感谢耶稣让我救赎自己。

Dan McGee 是目前生活在伊利诺伊州芝加哥的一名软件开发人员，主要在芝加哥地区全职做 Java Web 开发，并且有多年的经验；当然，他也使用其他语言。Dan 也从事一些自由项目。2007 年，Dan 成为 Arch Linux 发行版的一名开发者，并且从那以后做了很多和 Arch Linux 相关的项目，包括破解包管理器代码，兼职管理系统，以及帮助维持和改善网站。

Lawrence Oluyede 是一名 26 岁的 Python 和网络编程方面的软件开发专家。他非常乐见编程并行化以及函数式编程语言成为主流。他是由 *Apogeo* 出版的意大利第一本 Ruby 书（*Ruby per applicazioni web*）的意大利版的共同作者和审校。他在过去也为其他的书做过贡献，像 *Python Cookbook*（<http://www.amazon.com/Python-Cookbook-Alex-Martelli/dp/0596007973/>）以及 *The Definitive Guide to Django*（<http://www.amazon.com/Definitive-Guide-Django-Development-Right/dp/1590597257>）。

前言

这本书将向你介绍面向对象范型的术语,通过一步步的例子,专注于面向对象的设计。它将带你从简单的继承开始,这在面向对象程序员工具箱里是最有用的工具之一,到最复杂之一的合作继承。你将能提高、处理、定义以及操作异常。

你将能够把 Python 面向对象和不是那么面向对象的方面结合起来。通过研究高级的设计模式,你将能够创建可维护的应用程序。你将学习 Python 复杂的字符串和文件操作,以及如何区分二进制和文本数据。将会介绍给你不止一个,而是两个非常强大的自动化测试系统。你将会理解单元测试的喜悦以及创建单元测试是多么简单。你甚至会学习像数据库连接和 GUI 工具包这样的高级库,以及它们是如何应用面向对象原则的。

这本书讲了什么

第 1 章,面向对象设计覆盖了重要的面向对象概念。它主要处理关于抽象、类、封装和继承。在建模我们的类和对象时,我们也简要地看了下 UML。

第 2 章,Python 对象讨论了类和对象,以及它们是如何在 Python 中使用的。我们将学习 Python 对象中的属性和行为,以及把类组织成包和模块。最后我们将看到如何保护我们的数据。

第 3 章,当对象是相似的让我们从更深层次的视角来看继承。它覆盖了多重继承以及向我们展示了如何从内置来继承。这一章还包括了多态以及鸭子类型。

第 4 章,异常处理讲解异常和异常处理。我们将学习如何创建自己的异常。它还涉及了把异常用于程序流程控制。

第 5 章,何时使用面向对象编程主要处理对象,什么时候创建和使用它们。我们将看到如何使用属性来包装数据,以及限制数据访问。这一章也会讨论 DRY 原则以及如何不重复代码。

第 6 章,Python 数据结构覆盖了数据结构的面向对象特征。这一章主要处理元组、字典、列表和集合。我们也会看一看如何扩展内置对象。

第 7 章，Python 里面面向对象的快捷方式顾名思义，在 Python 中如何省时。我们将看到很多有用的内置函数，然后继续在列表、集合和字典中使用解析。我们将学习关于生成器、方法重载，以及默认参数的内容。我们还会看到如何把函数当成对象来用。

第 8 章，Python 设计模式 1 第一次向我们介绍了 Python 设计模式。然后我们将会看到装饰器模式、观察者模式、策略模式、状态模式、单例模式以及模板模式。这些模式将会和在 Python 中实现的合适的例子和程序一起讨论。

第 9 章，Python 设计模式 2 讲解上一章剩下的内容。我们将通过 Python 中的例子看到适配器模式、外观模式、享元模式、命令模式、抽象模式以及组合模式。

第 10 章，文件和字符串学习字符串和字符串格式化。也会讨论字节和字节数组。我们也将学习文件以及如何从文件读取数据和往文件里写数据。我们将学习存储和 pickle 对象，最后本章会讨论序列化对象。

第 11 章，测试面向对象的程序以使用测试和为什么测试如此重要开头。重点是测试驱动开发。我们将看到如何使用 unittest 模块，以及 py.test 自动化测试套件。最后我们将使用 coverage.py 来看代码的测试覆盖率。

第 12 章，常用 Python 3 库 集中介绍库以及它们在程序构建中的利用率。我们将使用 SQLAlchemy 来构建数据库，使用 TkInter 和 PyQt 开发用户界面。这一章会继续讨论如何构建 XML 文档以及我们应该如何使用 ElementTree 和 lxml。最后我们将使用 CherryPy 和 Jinja 来创建一个 Web 应用。

对于这本书你需要什么

为了编译运行在本书中提到的例子，你需要下面的软件：

- Python 3.0 或者更高的版本
- py.test
- coverage.py
- SQLAlchemy
- pygame
- PyQt
- CherryPy
- lxml

谁需要这本书

如果你是面向对象编程技术的新手，或者你有基本的 Python 技巧，并且希望深入地学习如何以及什么时候在 Python 中正确地应用面向对象编程，这本书适合你。

如果你是一个其他语言的面向对象编程人员，你也会发现这本书是对 Python 的一个有用的介绍，因为它使用了一些你已经熟悉的术语。

那些寻求迈入 Python 3 新世界的 Python 2 程序员也将会发现这本书的好处，但是其实你不需要了解 Python 2。

惯例

在这本书中，你将发现一些用于区分不同种类信息的文本风格。这里有一些这些风格的例子，并且介绍了它们的意义。

文本形式的代码会按照下面显示：“我们可以通过使用 `import` 语句来访问 Python 的其他模块”。

一个代码块会像下面这样显示：

```
class Friend(Contact):
    def __init__(self, name, email, phone):
        self.name = name
        self.email = email
        self.phone = phone
```

当我们希望你对一个代码片段里特殊的部分引起注意的时候，相应的行或者元素会设置成粗体：

```
class Friend(Contact):
    def __init__(self, name, email, phone):
        self.name = name
        self.email = email
        self.phone = phone
```

任何一个命令行的输入或输出都是下面这样的形式：

```
>>> e = EmailableContact("John Smith", "jsmith@example.net")
>>> Contact.all_contacts
```


新的术语或者重要的词将会显示成黑体。例如你在屏幕、菜单和对话框中看到的词，会以文本的形式这样显示：“你每一次点击 **Roll!** 按钮的时候，我们会通过这个特性来给标签更新一个新的随机值”。



警告或者重要的笔记将会这么显示。



小窍门和诀窍会这么显示。

读者反馈

我们随时欢迎来自读者的反馈信息。请告诉我们你对本书的评价——你喜欢或者不喜欢的地方。读者的反馈对于我们是非常重要的，这会帮助我们为你创作更有价值的作品。

请将你的反馈意见通过邮件 feedback@packtpub.com 发送给我们，并在邮件的主题中标明相关图书的名称。

如果你需要一本书并且希望我们出版相关图书，请在 www.packtpub.com 用 SUGGEST A TITLE 表格发一个纸条给我们或通过电子邮件 suggest@packtpub.com 告诉我们。

如果你具有关于某个主题的专业知识，并且有兴趣参与图书的编写，请查看我们的作者指南，网址：www.packtpub.com/authors。

客户支持

你现在是拥有 Packt 书的尊贵主人，购买本书我们会从以下方面帮助你。

下载本书的示例代码



你可以通过你在 <http://www.packtpub.com> 的账号来下载你购买图书的所有示例代码文件。如果你从其他地方购买的本书，你可以访问 <http://www.packtpub.com/support> 并且注册，我们会直接通过电子邮件发给你文件。

勘误

虽然我们已尽力确保我们内容的准确性，但是错误还是会发生。如果你发现本书的一个错误——文字或者代码错误——如果你能报告给我们，我们将非常感激。通过这样做，你可以把其他读者从挫败中拯救出来并且帮助我们改善这本书的后续版本。如果你发现任何错误，请通过访问 <http://packtpub.com/support> 来报告它们，选择你的书，点击 **let us know** 链接，并且输入错误的细节。一旦你的勘误被核实，你提交的信息将会被接收并且将会把勘误上传到我们的网站，或者在添加到一个已有的勘误列表的相应主题部分。从 <http://www.packtpub.com/support> 选择我们的信息标题将看到任何一个存在的勘误。

盗版

互联网上著作权侵害是在所有媒体中持续存在的问题。在 Packt，我们会非常重视对版权和许可证的保护。如果你在互联网上发现了对于我们著作的任何形式的任何非法复制，请立即告诉我们地址或者网站名字，以便我们能进行补救。

请通过 copyright@packtpub.com 联系我们，可附带一个可疑链接的盗版材料。

我们非常感谢你在保护我们的作者以及我们能带给你有价值内容的能力方面给予我们的帮助。

问题

如果你对这本书的任何方面有问题，你可以通过 questions@packtpub.com 联系我们，我们将尽力解决这个问题。

目录

第 1 章 面向对象设计	1
面向对象	1
对象和类	3
指定属性和行为	5
数据描述对象	5
行为是动作	7
隐藏细节并且创建公共接口	8
组合和继承	10
继承	12
案例学习	15
练习	22
总结	23
第 2 章 Python 对象	24
创建 Python 类	24
添加属性	26
让类实际做一些事情	26
对象的初始化	29
解释你自己	32
模块和包	34
组织模块	36
谁可以访问我的数据	41
案例学习	43
练习	52
总结	53

第 3 章 当对象是相似的	54
基本继承	54
扩展内置类	56
重写和 Super	58
多重继承	59
钻石的问题	61
不同的参数集合	66
多态	68
案例学习	71
练习	84
总结	85
第 4 章 异常处理	86
抛出异常	86
抛出一个异常	88
当一个异常产生时发生了什么	90
异常处理	91
异常层级	97
定义自己的异常	98
异常不是例外	99
案例学习	102
练习	112
总结	113
第 5 章 何时使用面向对象编程	114
把对象当作“对象”来对待	114
使用 property 为类中的数据添加行为	118
property 是怎样工作的	121
装饰器：创建 property 的另一种方法	123
何时该使用 property 属性	125
管理对象	127

移除重复的代码.....	130
实践一下.....	131
或者我们可以使用组合.....	135
案例学习.....	137
练习.....	145
总结.....	146
第6章 Python 数据结构.....	147
空对象.....	147
元组和命名元组.....	148
命名元组.....	150
字典.....	152
何时应该使用字典.....	155
使用 defaultdict.....	156
列表.....	157
对列表排序.....	160
集合.....	162
扩展内置数据类型.....	166
案例学习.....	171
练习.....	178
总结.....	178
第7章 Python 里面向对象的快捷方式.....	180
Python 内置函数.....	180
Len.....	180
Reversed.....	181
Enumerate.....	182
Zip.....	183
其他函数.....	185
解析.....	186
列表解析.....	186
集合和字典解析.....	188

生成器表达式.....	189
生成器.....	191
方法重载的另一种选择.....	194
默认参数.....	195
可变参数列表.....	197
参数拆分.....	201
函数也是对象.....	202
使用函数作为属性.....	206
可调用对象.....	207
案例学习.....	208
练习.....	212
总结.....	213
第 8 章 设计模式 1.....	214
设计模式.....	214
装饰器模式.....	215
装饰器实例.....	216
Python 中的装饰器模式.....	219
观察者模式.....	221
观察者实例.....	222
策略模式.....	224
策略实例.....	225
Python 中的策略模式.....	226
状态模式.....	227
状态实例.....	227
状态和策略模式的对比.....	233
单件模式.....	234
单件的实现方式.....	234
模块变量能够模仿单件.....	235
模板模式.....	238
模板实例.....	238

练习	242
总结	243
第 9 章 设计模式 2	244
适配器模式	244
外观模式	247
享元模式	250
命令模式	254
抽象工厂模式	259
组合模式	263
练习	267
总结	268
第 10 章 文件和字符串	270
字符串	270
字符串操作	271
字符串格式化	274
字符串是 Unicode 的	281
可变字节字符串	285
文件 I/O	286
把它放在上下文	287
伪造文件	289
存储对象	290
定制 pickle	292
序列化 Web 对象	294
练习	297
总结	299
第 11 章 测试面向对象的程序	300
为什么要测试	300
测试驱动开发	302
单元测试	303

断言方法.....	304
减少样板和清理.....	306
组织和运行测试.....	308
忽略失败的测试.....	309
用 py.test 测试.....	311
一个处理安装和清理的方法.....	313
一种完全不同的变量设置方式.....	316
用 py.test 跳过测试.....	320
py.test 的补充.....	321
多少测试才算够.....	323
案例学习.....	326
实现它.....	327
练习.....	332
总结.....	333
第 12 章 常用 Python 3 库.....	334
数据库访问.....	335
引入 SQLAlchemy.....	336
漂亮的用户界面.....	340
TkInter.....	341
PyQt.....	345
选择一个 GUI 工具包.....	347
XML.....	348
ElementTree.....	349
Lxml.....	353
CherryPy.....	354
一个完整的 Web 堆栈.....	357
练习.....	363
总结.....	364

1

第 1 章 面向对象设计

在软件开发过程中，通常认为在编程之前要完成设计。这种说法并不准确；在实际中，分析、编程和设计趋向于重叠、结合和交织。通过本章内容，我们会学到：

- 面向对象是什么意思。
- 面向对象设计和面向对象编程的不同。
- 面向对象设计的基本原则。
- 基本统一建模语言并且当它不是一个麻烦的时候。

面向对象

所有人都知道对象是什么：某种我们可以感知、触摸和操纵的有形的东西。我们接触到的最早的对象是典型的婴儿玩具。木质积木、塑料形体及过量的拼图都是常见的第一对象。婴儿可以很快地学习到，特定的对象可以做特定的事情。三角形物体能准确地放到三角形的孔里。铃可以响、按钮可以按，以及手柄可以扳。

在软件开发中，对象的定义也没有太大的差别。对象通常不是你可以捡起、感知或者触摸的有形的东西，而是这些东西的模型，这些模型可以做特定的事情并且特定的事情可以作用于它们身上。正式地讲，对象是一个数据以及相关行为的集合。

现在我们知道了什么是对象，那么什么是面向对象呢？面向简单的意思就是指向。所以面向对象简单的意思就是“功能上指向建模对象”。通过数据和行为来描述交互对象的集合，这是用于对复杂系统建模的众多技术之一。

如果你读过任何的宣传资料，那么你可能已经碰到了面向对象分析、面向对象设计、