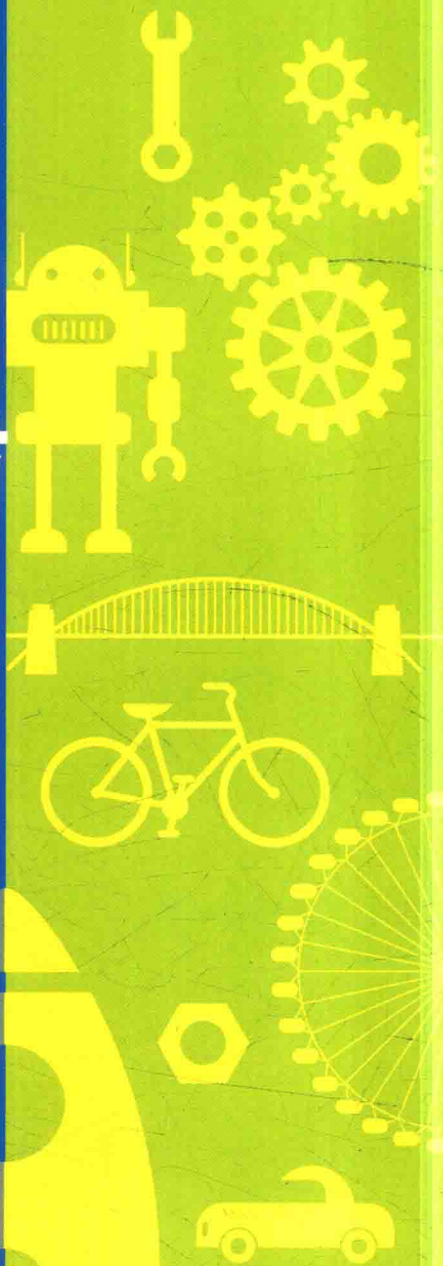




Begin to Code with Python

学习编程第一步

零基础上手 Python开发



作者: 罗布·麦尔斯(Rob Miles) 著
译者: 曹瑜 郭俊凤 译



清华大学出版社

学习编程第一步

零基础上手 Python 开发

[英] 罗伯·麦尔斯(Rob Miles) 著
曹瑜 郭俊凤 译



清华大学出版社

北京

Authorized translation from the English language edition, entitled BEGIN TO CODE WITH PYTHON, 1st Edition, 9781509304523 by MILES, ROB, published by Pearson Education, Inc, publishing as Microsoft Press, Copyright © 2018.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc. CHINESE SIMPLIFIED language edition published by TSINGHUA UNIVERSITY PRESS LIMITED, Copyright © 2018.

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

学习编程第一步 零基础上手 Python 开发 / (英) 罗伯·麦尔斯(Rob Miles) 著; 曹瑜, 郭俊凤 译. —北京: 清华大学出版社, 2018

书名原文: Begin to Code with Python

ISBN 978-7-302-50249-4

I. ①学… II. ①罗… ②曹… ③郭… 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(2018)第 103231 号

责任编辑: 王 军 于 平

封面设计: 牛艳敏

版式设计: 思创景点

责任校对: 曹 阳

责任印制: 刘海龙

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 三河市君旺印务有限公司

经 销: 全国新华书店

开 本: 170mm×240mm 印 张: 31.25 字 数: 630 千字

版 次: 2018 年 6 月第 1 版 印 次: 2018 年 6 月第 1 次印刷

印 数: 1~4000

定 价: 79.80 元

产品编号: 078950-01

译者序

Python 是一种解释型、面向对象、动态数据类型的高级程序设计语言。通过 Python 编程，我们能解决现实生活中的很多问题。自 1991 年问世以来，Python 日益流行，现已成为多个国家计算机教学中的首选入门编程语言之一。Python 虽然简单，其设计却十分严谨。Python 可能没有 C 或 C++ 这类编译型语言运行速度那么快，但是 C 和 C++ 需要你无时无刻地关注数据类型、内存溢出、边界检查等问题；Python 就像一个贴心的仆人，私下为你处理好这些，而你可将全部心思放在程序的设计逻辑上。

本书是一本编程启蒙书籍，将为你的编程之旅指明方向，指引你朝着正确的、有用的目标前进。本书共分三部分。第 I 部分介绍如何安装和使用编程工具，讲述 Python 编程语言的基本元素。第 II 部分描述如何利用 Python 编程语言的特点构建复杂的应用程序；指导你设计、测试 Python 应用程序，并为程序编写文档。第 III 部分介绍如何使用软件库在图形用户界面中创建应用程序，如何在网络应用中将 Python 程序用作客户端和服务端。最后创建充满魅力的游戏。

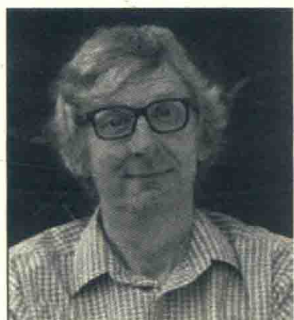
即使你没有编程经验，也可在本书的指导下编写出解决实际问题的软件。本书完全颠覆了编程入门书籍，紧跟时代，为当今的编程新手提供深入的讲解，令其他书籍望尘莫及。本书编排新颖，从“代码分析”“易错之处”“动手练习”到“专家点津”，无不令人耳目一新。无论你是初出茅庐的新手，还是经验丰富的编程人员，本书都将赋予你能量、激动和快乐！当你看到其他用户乐于使用你写的程序做一些有用的事情时，你的内心将是十分甜蜜的。

在这里要感谢清华大学出版社的编辑们，他们为本书的翻译投入了巨大热情并付出了很多心血。没有他们的帮助和鼓励，本书不可能顺利付梓。翻译过程中，译者力求翻译表达传神，但是鉴于水平有限，难免出现错误和失误，请不吝指正。感激不尽！

本书主要内容由曹瑜、郭俊凤翻译，参与本次翻译的还有陈妍、何美英、陈宏波、熊晓磊、管兆昶、潘洪荣、曹汉鸣、高娟妮、王燕、谢李君、李珍珍、王璐、王华健、柳松洋、曹晓松、陈彬、洪妍、刘芸、邱培强、高维杰、张素英、颜灵佳、方峻、顾永湘、孔祥亮，在此一并表示感谢！

最后，希望读者在阅读过程中领略到 Python 语言之美！

作者简介



Rob Miles 在英国赫尔大学讲授编程课程的年限超过 30 年。他是一位 Microsoft MVP，对编写程序和创造新事物充满热情。即使在空闲时间，也不忘继续编写代码。他喜欢编写代码，然后运行这些代码，看看会有什么结果。他认为编程最富有创意，让你可以一探究竟。他相信，编程可以构筑未来。

Rob 自诩为一个段子手，但旁人却从未亲耳听到他讲的笑话。如果你想进一步接触和了解 Rob，则可访问他的博客 www.robmiles.com，或通过 @RobMiles 在 Twitter 上找到他。

前 言

要对事物一探究竟，编程是最富有创意的活动。为什么这样讲呢？如果你学习绘画，你可以绘出作品。如果你学习拉小提琴，你能演奏出音乐。但如果你学习编程，那将是一番新体验(当然，如果愿意，也可编写代码来绘画或演奏)。一旦踏上编程之旅，你去往哪里将再无限制。新的设备、技术和市场层出不穷，你的编程技能总有用武之地。

本书是一本编程启蒙书籍，将为你的编程之旅指明方向，指引你朝着正确的、有用的目标前进。通过学习本书，你将掌握编程技能，并编写出有用的程序。

不过，在出发前，先要声明几个注意点。就像游猎时，导游会向你介绍狮子、老虎和鳄鱼，我要向你透露一点，我们的旅程并非是一帆风顺的。对于如何解决问题，编程人员必须稍微转变一下思路，原因是计算机的工作方式与人类不同。人类可以小心翼翼地完成复杂事情，计算机可极快地完成简单工作。编程人员的职责是利用计算机的简单能力去解决复杂问题；这就是我们要学的内容。编程人员取得成功的关键与其他很多职业是类似的。要成为一位闻名世界的小提琴家，就必须做大量的练习。编程同样如此，只有耗费大量时间处理程序，才能掌握编程技能。当你看到其他用户乐于使用你编写的程序做一些有用的事情时，你的内心将是十分甜蜜的。

本书的编排方式

本书共分三部分。后一部分建立在前一部分的基础上，以便循序渐进地引导你成为一名成功的编程人员。首先讲述基本编程指令，这些指令将告知计算机如何完成任务。最后将讲解如何构建专业软件。

第 I 部分 编程基础知识

第 I 部分介绍如何安装和使用编程工具，讲述 Python 编程语言的基本元素。你将学会编写第一个程序，领会所有程序的底层构建方式。你将了解 Python 在编程语言中的地位 and 重要性。

第 II 部分 高级编程

第 II 部分描述如何利用 Python 编程语言的特点来构建复杂的应用程序。从中你将了解到，可将大程序分解为多个较小的元素，也可根据要解决的具体问题创建自定义数据类型；你将能设计、测试 Python 应用程序，并为程序编写文档。

第 III 部分 Python 实战

学会如何自己编写程序后，下一步学习如何使用其他人编写的代码。Python 的一个重要优势在于：Python 用户可使用大量的软件库。第 III 部分将介绍其中的很多软件库，并讲述如何使用这些库在图形用户界面中创建应用程序，如何在网络应用中将 Python 程序用作客户端和服务端。最后将创建充满魅力的游戏。

可将本书第 III 部分作为一个可下载的文档，你可在桌面上打开该文档，体验所演示的程序，并自行创建应用程序。

学习本书的方式

每一章都讲解有关 Python 编程的新知识，先讲解如何做，然后指导你动手练习并自行创建程序。在这本精心撰写的书籍中，没有哪一页是多余的。不过，你也可根据自己的需要选学。

你可从头到尾阅读本书，这样，你可慢慢消化更多知识，直至编写能够使用的程序。就像骑自行车一样，从做中学。为此，你必须投入时间，不断练习。本书将指导你掌握知识，使你信心百倍地着手编程；如果此前你的程序未能如期运行，你也可从本书中寻找灵感。为此，本书纳入了以下一些特殊段落。

动手练习

没错，从做中学效果最好。全书穿插多个“动手练习”，带你亲手练习编程技能。每个“动手练习”都首先引入一个示例，然后列出一些你可以自己去尝试的步骤。创建的代码能在 Windows、Mac OS 或 Linux 上运行。

代码分析

学习编程的一种绝佳方式是分析其他人编写的代码，并分析其工作原理，有时需要查明为什么代码不能如期工作。本书包含 150 多个可供你分析的示例程序。你将推理论证程序的行为、改正错误，并提供改进建议。

易错之处

如果你此前从未听说过程序会失败，那么在编写你的第一个程序之后，你很快将体验到酸楚。为帮助你处理问题，本书添加了“易错之处”特殊段落，以推测你可能遇到的问题，并提出解决方案。例如，在介绍新知识点时，总会讨论哪些地方容易出错；在讨论新功能时，总是告诉你一些需要考虑的事项。

专家点津

我讲授编程课程已 30 年有余，也编写过很多商业程序。我知道，开始时必须掌握一些要点。“专家点津”预先为你呈现这些信息，使你在学习过程中，可以像专业开发人员那样考虑问题。其中涵盖的问题十分广泛，从编程到工作态度，再到哲理。强烈建议你阅读领会这些观点，这样，在未来可节省大量时间！

软件和硬件要求

为练习本书中的程序，你需要配备一台计算机，并安装一些软件。我恐怕没法给你提供计算机，但在第 1 章中，你可以了解到从何处获得 Python 工具和 Visual Studio Code 应用程序；开始创建更大型的应用程序时，这些都必不可少。

使用 PC 或笔记本电脑

可使用 Windows、Mac OS 或 Linux 来创建和运行本书中的 Python 程序。对 PC 硬件的最低要求如下：

- 1GHz 或主频更高的处理器，最好使用 Intel i5 或更高性能的处理器。
- 至少4GB内存，最好是8GB内存。
- 256GB硬件空间(完整安装Python和Visual Studio Code将占用约1GB的硬盘空间)。

对于显卡没有具体要求，但编写程序时，更清晰的屏幕当然会好一些。

移动设备

可在移动设备或平板电脑上编写和运行 Python 程序。如果有一台运行 iOS 的 Apple 设备，建议你使用 Pythonista 应用。如果使用 Android 设备，应使用 Pyonic Python 3 解释器。

使用 Raspberry Pi

如果手头缺少资金，开始时可使用最廉价的 Raspberry Pi(运行 Raspbian 操作系统)，其中内置 Python 3.6 和 IDLE 开发环境。

源代码下载

本书每一章都演示和解释程序，以帮助你理解编程机理。你也可在这些程序的基础上自行编写其他程序，可访问 <https://aka.ms/BeginCodePython/downloads>，或扫描本书封底的二维码，下载本书的示例代码。

可按第 1 章的说明来安装示例程序和代码。

致 谢

感谢 Laura Norman 给予我撰写本书的机会，感谢 Dan Foster 和 Rick Kughen 对本书的整理和编辑，感谢 John Ray 富有建设性的精辟技术建议，感谢 Tracey Croom、Becky Winter 和 Rob Nance 的排版和加工。

最后感谢妻子 Mary 的支持，你为我沏好一杯又一杯清茶，还为我烘焙软饼，谢谢你的爱。

勘误、更新和相关支持

我们已经尽最大努力，确保本书的信息正确无误。可访问 <https://aka.ms/BegintoCodePython/errata> 来获得本书的更新信息，其中包括已提交的勘误和相关的更正意见。

如果你发现了未在勘误表中列出的其他错误，也可将其提供给以上网页。

如果需要其他支持，可给 Microsoft Press Book Support 发送电子邮件：mspinput@microsoft.com。

请注意，上述地址不提供 Microsoft 软件和硬件支持。如果需要获得有关 Microsoft 软件和硬件的帮助，可访问 <http://support.microsoft.com>。

获得 MTA 认证

MTA(Microsoft Certified Professional)计划旨在使你获得技术认证。通过 98-381 考试“Introduction to Programming Using Python”，你将获得 MTA 的 Python 编程认证，可通过以下网址了解考试详情：<https://www.microsoft.com/en-us/learning/exam-98-381.aspx>。

为帮助你尽量通过学习本书获得 MTA 认证，这里将考点与本书素材一一对应列出。注意，考点基于 2017 年 10 月公布的考试大纲。

认证结构

将认证划分为多个主题，每个主题都占一定比例。你可以阅读每个主题，并找到与本书对应的素材。下面的表格列出了对应关系。

1) 使用数据类型和运算符执行操作(20%~25%)

该主题与以下几章的关系最密切：第 4 章描述 Python 程序的数据处理要点，以及如何操纵和存储文本和数值；第 5 章介绍布尔变量类型，讲述如何操作逻辑值；第 8 章描述如何存储数据集合；第 9 章讲述如何创建和存储数据结构；第 11 章详述 Python 程序中的集合。

考点	本书的对应章节
表达式求值，识别 Python 为每个变量指定的数据类型。识别 str、int、float 和 bool 数据类型	第 4 章：Python 中的变量 第 4 章：使用文本 第 4 章：使用数值 第 5 章：布尔数据 第 5 章：布尔表达式

(续表)

考点	本书的对应章节
执行数据和数据类型操作。将一种数据类型转换为另一种；构建数据结构；执行索引和切片操作	第 4 章：将字符串转换为浮点值 第 4 章：在浮点数和整数之间转换 第 4 章：实数和浮点数 第 8 章：列表和销售金额跟踪 第 9 章：使用类存储联系人信息
根据运算符的优先级确定执行顺序；赋值运算符(=)、比较运算符、逻辑运算符、算术运算符、身份运算符(is)、成员运算符(in)	第 4 章：执行计算 第 9 章：Contact 对象和引用 第 11 章：集合和标记
选择适当的运算符取得预期结果：赋值运算符、比较运算符、逻辑运算符、算术运算符、身份运算符(is)、成员运算符(in)	第 4 章：执行计算 第 9 章：布尔表达式

2) 使用决策和循环控制流程(25%~30%)

该主题与以下几章的关系最密切：第 5 章介绍 if 结构；第 6 章讲述用于实现 Python 程序循环的 while 和 for 循环结构。

考点	本书的对应章节
构建和分析使用分支语句的代码段	第 5 章：布尔数据 第 5 章：if 结构
if、elif、else、嵌套和复合条件表达式	第 5 章：if 结构
构建和分析执行迭代的代码段	第 6 章：while 结构 第 6 章：for 循环结构
while、for、break、continue、pass、嵌套的循环以及包含复合条件表达式的循环	第 6 章：while 结构 第 6 章：for 循环结构 第 8 章：使用冒泡排序法

3) 执行输入和输出操作(20%~25%)

从第 3 章和第 4 章描述的第一个程序开始，本书穿插介绍如何使用控制台输入和输出函数。第 8 章简要介绍在 Python 程序中使用文件存储，第 9 章进一步介绍如何使用 Python pickle 库将数据结构保存在文件中。第 10 章详细介绍 Python 程序中的字符串格式化工具。

考点	本书的对应章节
构建和分析执行文件输入和输出操作的代码段。理解 open、close、read、write、append、check existence、delete、with 语句	第 8 章：在文件中保存数据 第 9 章：使用 pickle 将联系人信息保存到文件中

(续表)

考点	本书的对应章节
构建和分析执行控制台输入和输出操作的代码段；包括从控制台读取、打印带格式文本以及使用命令行参数	第 3 章：使用 print 函数获取程序输出 第 4 章：使用 input 函数读取文本 第 10 章：设置 Python 字符串格式

4) 代码的文档化和结构(15%~20%)

本书强调：代码的结构必须清晰，代码必须进行文档化。第 3 章简要介绍 Python 注释，第 5 章强调良好代码布局的重要性。第 7 章讲述通过 Python 函数完善程序结构，并描述如何给函数添加文档，作为程序的自我说明。

考点	本书的对应章节
使用注释和文档字符串实现代码段的文档化。使用缩进、空白、注释和文档字符串；使用 pydoc 生成文档	第 3 章：Python 注释 第 5 章：缩进文本可导致大问题 第 7 章：为函数添加帮助信息 第 12 章：查看程序文档
构建和分析包含函数定义的代码段：调用签名、默认值、return、def、pass	第 7 章：函数的组成

5) 故障排除和错误处理(5%~10%)

第 3 章介绍 Python 代码中的语法错误。第 4 章描述数据处理中的运行时错误。第 6 章和第 7 章讨论应用程序开发环境中产生逻辑错误的原因和后果。第 6 章和第 10 章讲述 Python 程序如何抛出和管理异常。第 12 章讲述在 Python 程序测试中使用单元测试。

考点	本书的对应章节
分析、检测和纠正包含以下错误的代码段：语法错误、逻辑错误、运行时错误	第 3 章：不完整的程序 第 4 章：输入错误和测试 第 5 章：相等和浮点值 第 6 章：看似正确的循环出错了 第 7 章：使用调试器探究程序 第 12 章：程序测试
分析和构建用于处理异常的代码段：try、except、else、finally、raise	第 6 章：使用异常检测无效编号输入 第 10 章：抛出异常指示错误

6) 使用模块和工具执行操作(1%~5%)

本书使用的很多 Python 模块都以 random 和 time 模块开头。第 13 章使用 random 库的函数来创建 random 工件，第 16 章的时间跟踪应用程序使用 time 库的函数。

考点	本书的对应章节
使用内置模块执行基本操作: math、datetime、io、sys、os、os.path、random	第 3 章: random 库 第 3 章: time 库
使用内置模块解决复杂计算问题: math、datetime、random	第 10 章: Time Tracker 中的会话跟踪 第 16 章: 生成工件

联系我们

Microsoft Press 将“让读者满意”奉为至高无上的宗旨，你的反馈是我们最宝贵的资产。请通过 <https://aka.ms/tellpress> 将你对本书的意见发给我们。

为节省时间，其中只有几个简短问题。你的回答将直接反馈到 Microsoft Press(不需要提供个人信息)。

提前谢谢你的反馈!

另外，你可通过 Twitter(<http://twitter.com/MicrosoftPress>)与我们交流。

目 录

第 I 部分 编程基础知识

第 1 章 开始使用 Python	3
1.1 什么是 Python	3
1.1.1 Python 的起源	4
1.1.2 Python 版本	4
1.2 打造使用 Python 的场所	5
1.2.1 获取工具	5
1.2.2 Python(适用于 Windows PC)	6
1.3 启动 Python	8
1.4 本章小结	11
第 2 章 Python 和编程	13
2.1 编程人员的工作	13
2.1.1 编程和规划生日聚会	13
2.1.2 编程和问题	14
2.1.3 编程人员要善于沟通	15
2.2 计算机用于处理数据	16
2.2.1 机器、计算机和人	16
2.2.2 程序用于处理数据	18
2.2.3 Python 用于处理数据	19
2.3 数据和信息	23
2.4 使用 Python 函数	27
2.4.1 ord 函数	27
2.4.2 chr 函数	28
2.4.3 使用 bin 函数探索数据 存储	29
2.5 本章小结	30

第 3 章 Python 程序结构	33
3.1 编写首个 Python 程序	33
3.1.1 使用 IDLE 运行 Python 程序	33
3.1.2 使用 print 函数获得程序 输出	37
3.2 使用 Python 库	42
3.2.1 random 库	42
3.2.2 time 库	45
3.3 Python 注释	46
3.4 从桌面运行 Python	47
3.5 添加一些 snaps 函数	48
3.5.1 添加 pygame 库	48
3.5.2 snaps 函数	49
3.6 本章小结	52
第 4 章 使用变量	55
4.1 Python 中的变量	55
4.2 使用文本	59
4.2.1 标记字符串的开始和 结束	61
4.2.2 文本中的转义字符	62
4.2.3 使用 input 函数读取 文本	63
4.3 使用数值	65
4.3.1 将字符串转换为整 型值	65
4.3.2 整数和实数	66
4.3.3 实数和浮点数	67
4.3.4 将字符串转换为浮 点值	71

4.3.5	执行计算	72	6.2	for 循环结构	120
4.3.6	在浮点数和整数之间 转换	73	6.3	使用 snaps 制作数字 闹钟	123
4.4	snaps 函数 get_weather_ temp	75	6.4	本章小结	124
4.5	本章小结	76	第 7 章	使用函数简化程序	125
第 5 章	在程序中制定决策	79	7.1	函数的组成	125
5.1	布尔数据	79	7.1.1	使用参数给函数提供 信息	128
5.1.1	创建布尔变量	79	7.1.2	函数调用的返回值	135
5.1.2	布尔表达式	81	7.2	构建可重用的函数	141
5.1.3	比较值	83	7.2.1	创建文本输入函数	141
5.1.4	布尔运算	86	7.2.2	为函数添加帮助信息	142
5.2	if 结构	88	7.2.3	创建数值输入函数	144
5.2.1	嵌套 if 条件	95	7.2.4	将函数转换为 Python 模块	147
5.2.2	使用逻辑	96	7.2.5	使用 IDLE 调试器	148
5.3	使用决策创建应用程序	96	7.3	本章小结	153
5.3.1	设计用户界面	97	第 8 章	存储数据集合	155
5.3.2	实现用户界面	98	8.1	列表和销售金额跟踪	155
5.3.3	测试用户输入	99	8.1.1	单独变量的局限性	157
5.3.4	完成程序	99	8.1.2	Python 中的列表	158
5.4	导入 snaps 库	100	8.1.3	在列表中读取值	160
5.5	本章小结	103	8.1.4	使用 for 循环显示 列表	161
第 6 章	使用循环执行重复 操作	105	8.2	将程序重构为函数	162
6.1	while 结构	105	8.2.1	创建占位函数	165
6.1.1	使用 while 重复执行一 系列语句	105	8.2.2	创建用户菜单	165
6.1.2	处理无效的用户输入	109	8.3	使用冒泡排序法	167
6.1.3	使用异常检查无效编号 输入	112	8.3.1	使用测试数据初始化 列表	167
6.1.4	异常和编号读取	114	8.3.2	按降序排序列表	168
6.1.5	处理多个异常	115	8.3.3	按升序排序列表	172
6.1.6	跳出循环	116	8.3.4	查找最大值和最 小值	173
6.1.7	使用 continue 返回到循环 顶部	117	8.3.5	计算销售总额和平均 销售额	174
6.1.8	重复循环计数	118			

8.3.6	完成程序	174
8.4	在文件中存储数据	175
8.4.1	写入文件	176
8.4.2	写入销售金额	179
8.4.3	从文件读取	180
8.4.4	读取销售金额	182
8.4.5	处理文件错误	183
8.5	存储数据表	185
8.6	将列表用作查找表	188
8.7	元组	189
8.8	本章小结	191

第 II 部分 高级编程

第 9 章	使用类存储数据	197
9.1	生成 Tiny Contacts 应用	197
9.1.1	生成原型	198
9.1.2	在不同的列表中存储联系人详细信息	200
9.1.3	使用类存储联系人详细信息	202
9.1.4	在 Tiny Contacts 程序中使用 Contact 类	205
9.1.5	编辑联系人信息	207
9.1.6	使用 pickle 将联系人信息保存到文件中	215
9.1.7	使用 pickle 从文件加载联系人信息	217
9.1.8	给 Tiny Contacts 程序添加 save 和 load 函数	218
9.1.9	设置类实例	219
9.2	字典	223
9.2.1	字典管理	225
9.2.2	从函数返回字典	225
9.2.3	使用字典存储联系人信息	226

9.3	本章小结	227
第 10 章	使用类创建活动对象	229
10.1	创建 Time Tracker	229
10.1.1	为类添加数据特性	230
10.1.2	创建聚合对象	231
10.1.3	为类创建方法特性	232
10.1.4	为方法添加验证代码	234
10.1.5	防止数据特性受到破坏	243
10.1.6	受保护的方法	246
10.2	创建类属性	246
10.3	扩展类设计	250
10.4	类中的 __str__ 方法	257
10.5	Time Tracker 中的法庭工作时长跟踪	260
10.5.1	Python map 函数	264
10.5.2	Python join 方法	269
10.6	使用 snaps 制作音乐	270
10.7	本章小结	273
第 11 章	设计面向对象的解决方案	277
11.1	Fashion Shop 应用程序	277
11.1.1	应用程序数据设计	279
11.1.2	面向对象的设计	279
11.1.3	创建超类和子类	282
11.1.4	数据设计总结	295
11.1.5	实现应用程序行为	302
11.1.6	将对象作为组件	305
11.2	创建 FashionShop 组件	306

11.3	设计类	314
11.4	Python 集合	315
11.4.1	集合和标签	318
11.4.2	集合与类层次结构的对比	321
11.5	本章小结	324
第 12 章	Python 应用程序	327
12.1	高级函数	327
12.1.1	函数引用	327
12.1.2	使用 lambda 表达式	332
12.1.3	迭代器函数和 yield 语句	335
12.1.4	使用任意实参数量的函数	340
12.2	模块和包	343
12.2.1	Python 模块	343
12.2.2	给 BTCInput 添加 readme 函数	343
12.2.3	将模块作为程序运行	344
12.2.4	检测模块是否作为程序执行	344
12.2.5	创建 Python 包	345
12.2.6	导入包中的模块	347
12.3	程序测试	350
12.3.1	Python assert 语句	351
12.3.2	Python unittest 模块	352
12.3.3	创建测试	355
12.4	查看程序文档	356
12.5	本章小结	361

第 III 部分 Python 实战

第 13 章	Python 和图形用户界面	365
---------------	-----------------------	------------

13.1	Visual Studio Code	365
13.1.1	安装 Visual Studio Code	366
13.1.2	在 Visual Studio Code 中安装 Python Extension	366
13.1.3	创建项目文件夹	367
13.1.4	创建程序文件	368
13.1.5	调试程序	369
13.1.6	其他 Python 编辑器	373
13.2	使用 Tkinter 创建图形用户界面	373
13.2.1	创建图形应用程序	379
13.2.2	网格布局	380
13.2.3	创建事件处理器	382
13.2.4	创建 mainloop	383
13.2.5	处理 GUI 中的错误	383
13.2.6	显示消息框	385
13.2.7	在画布上绘制	388
13.2.8	Tkinter 事件	391
13.2.9	创建绘制程序	392
13.2.10	输入多行文本	394
13.2.11	在 Frame 中组合显示元素	396
13.2.12	使用 GUI 创建可编辑的 StockItem	396
13.2.13	创建一个 Listbox 选择器	402
13.2.14	包含 GUI 的应用程序	408
13.3	本章小结	410

第 14 章	Python 程序用作网络客户端	413
14.1	计算机网络	413

- 14.2 从 Python 使用 Web422
 - 14.2.1 读取网页 422
 - 14.2.2 使用基于 Web 的
数据 423
- 14.3 本章小结426
- 第 15 章 Python 程序用作网络
服务器..... 429**
 - 15.1 在 Python 中创建 Web
服务器.....429
 - 15.1.1 一台基于套接字的
小型服务器..... 429
 - 15.1.2 Python Web 服务器... 433
 - 15.1.3 通过文件支持
网页 435
 - 15.1.4 从 Web 用户获取
信息 439
 - 15.2 在 Web 上托管 Python
应用程序.....443
 - 15.3 本章小结443
- 第 16 章 使用 pygame 创建
游戏 445**
 - 16.1 开始使用 pygame.....445
 - 16.2 使用 pygame 绘图.....451
 - 16.2.1 图像文件类型451
 - 16.2.2 将图像加载到
游戏中452
 - 16.2.3 使图像能够移动.....454
 - 16.3 从 pygame 获取用户
输入.....455
 - 16.4 创建游戏 Sprite.....457
 - 16.4.1 添加玩家 Sprite.....461
 - 16.4.2 控制玩家 Sprite.....463
 - 16.4.3 添加 Cracker Sprite.....464
 - 16.4.4 添加大量 Sprite
实例465
 - 16.4.5 捕获薄脆饼干466
 - 16.4.6 添加西红柿杀手.....469
 - 16.5 完成游戏472
 - 16.5.1 添加启动屏幕472
 - 16.5.2 结束游戏475
 - 16.5.3 游戏记分476
 - 16.6 本章小结477