

SAMS
**Teach
Yourself**

- 全球销量逾百万册的系列图书
- 连续十余年打造的经典品牌
- 直观、循序渐进的学习教程
- 掌握关键知识的最佳起点
- “Read Less, Do More”（精读多练）的教学理念
- 以示例引导读者完成最常见的任务

每章内容针对初学者精心设计，**1**小时轻松阅读学习，
24小时彻底掌握关键知识

每章**案例与练习题**助你轻松完成常见任务，
通过**实践**提高应用技能，巩固所学知识

树莓派Python编程

入门与实战

[美] Richard Blum 著
Christine Bresnahan
王超 马立新 译



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

第1章 入门与实战

第2章 入门与实战

第3章 入门与实战

第4章 入门与实战

第5章 入门与实战

第6章 入门与实战

第7章 入门与实战

第8章 入门与实战

第9章 入门与实战

第10章 入门与实战

第11章 入门与实战

第12章 入门与实战

第13章 入门与实战



树莓派Python编程 入门与实战

[美] Richard Blum 著
Christine Bresnahan
王超 马立新 译

人民邮电出版社
北京

图书在版编目 (CIP) 数据

树莓派Python编程入门与实战 / (美) 勃鲁姆
(Blum, R.), (美) 布莱斯纳罕 (Bresnahan, C.) 著; 王
超, 马立新译. — 北京: 人民邮电出版社, 2015. 10
ISBN 978-7-115-39118-6

I. ①树… II. ①勃… ②布… ③王… ④马… 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167190号

版权声明

Richard Blum Christine Bresnahan: Sams Teach Yourself Python Programming for Raspberry Pi 24 Hours
ISBN: 978-0-7897-5205-5

Copyright © 2014 by Sams Publishing.

Authorized translation from the English language edition published by Sams.

All rights reserved.

本书中文简体字版由美国 Sams 出版公司授权人民邮电出版社出版。未经出版者书面许可, 对本书任何部分不得以任何方式复制或抄袭。

版权所有, 侵权必究。

内 容 提 要

树莓派是一个只有信用卡大小的裸露电路板, 它也是一个运行开源 Linux 操作系统的完全可编程的 PC 系统。树莓派的官方编程语言是 Python, 本书介绍了树莓派的 Python 编程方法。

本书共分 7 个部分。前 6 个部分介绍了树莓派编程环境、Python 基础知识、高级 Python、图形编程、业务编程和树莓派 Python 项目; 第 7 部分通过附录介绍了如何将树莓派操作系统加载到 Raspbian 卡上。

本书适合对 Python 编程感兴趣的读者和树莓派爱好者, 也适合想要基于低成本 Linux 平台开发应用的用户。

-
- ◆ 著 [美] Richard Blum Christine Bresnahan
 - 译 王 超 马立新
 - 责任编辑 陈冀康
 - 责任印制 张佳莹 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>
北京昌平百善印刷厂印刷
 - ◆ 开本: 787×1092 1/16
印张: 28
字数: 696 千字 2015 年 10 月第 1 版
印数: 1-2 500 册 2015 年 10 月北京第 1 次印刷
著作权合同登记号 图字: 01-2013-8453 号
-

定价: 69.00 元

读者服务热线: (010)81055410 印装质量热线: (010)81055316
反盗版热线: (010)81055315

作者简介

Richard Blum 作为网络和系统管理员已经工作了 25 年多，他曾经管理过超过 3500 个用户的 Microsoft、UNIX、Linux 和 Novell 服务器网络。他开发了编程和 Linux 课程，并通过网络教授给世界范围内的学院和大学的学生们。Rich 有 Purdue 大学的管理信息系统的硕士学位并且他是很多 Linux 书籍的作者，包括 Linux Command Line and shell Scripting Bible（与 Christine Bresnahan 为共同作者，2011，Wiley）、Linux for Dummies，第 9 版（2009，Wiley）、以及 Professional Linux Programming（与 Jon Master 为共同作者，2007，Wiley）。当他不再忙着变成一个计算机呆子时，他很享受和他的妻子 Barbara 以及两个女儿 Katie Jane 和 Jessica 在一起的生活。

作为一名系统管理员，Christine Bresnahan 在 IT 工业工作超过了 25 年。Christine 现在是印第安纳州印第安纳波利斯的常青藤技术社区学院的助理教授，她讲授 Python 编程、Linux 系统管理以及 Linux 安全课程。Christine 创作了一些 UNIX/Linux 的教学材料并且她是 Linux Bible 第 8 版的作者（与 Christopher Negus 为共同作者，2012，Wiley）以及 Linux Command Line and shell Scripting Bible 的作者（于 Richard Blum 为共同作者，2011，Wiley）。她从 2012 年变成一个树莓派的狂热分子。

我们期待你的反馈

作为本书的读者，你是我们最重要的批评家和评论家。我们很看重你的意见，想要知道什么我们做得好，我们能把什么做得更好，你想我们再出版什么其他内容，以及其他你想对我们说的。

我们欢迎你的评论。你可以发电子邮件或者写信让我们知道你喜不喜欢本书以及我们能做什么让我们的书变得更好。

请注意我们不能帮助你解决本书涉及的技术问题。

当你写信的时候，请确定包含了书籍的名称和作者以及你的名字和电子邮件地址。我们会非常小心地查验你的评论并且分享给在本书的作者以及在本书上工作过的编辑。

电子邮件：consumer@sampublishing.com

邮 箱：Sams Publishing

ATTN: Reader Feedback
800 East 96th Street
Indianapolis, IN 46240 USA

读者服务

访问我们的网站并在 informit.com/register 注册本书来方便地获取本书的更新、下载或者勘误表。

致 谢

非常感谢 Sams 出版社的优秀团队在这个项目上杰出的工作。感谢执行编辑 Rich Kughen，为我们提供了写作这本书的机会以及持续跟踪每一件事。我们非常感激开发编辑 Mark Renfrow，他的勤劳工作让我们的书更有变现力。感谢产品编辑 Andy Beaster，让我们确定这本会被出版。非常感谢文本编辑 Kitty Wilson，感谢她非常耐心和勤勉地让我们的内容更有可读性。同样，我们的技术编辑 Jason Foster 给了我们很大的帮助，他花费了很长的时间帮助我们仔细检查我们的工作并且保证书籍的技术准确。

感谢 Tonya Witting 摄影的 Tonya，他为我们的树莓派拍摄了那些不可思议的照片并且非常耐心地拍摄本书所需要的照片。我们也想要感谢 Waterside Productions, Inc. 的 Carole Jelen，他帮助我们安排了这次机会并且在我们的写作过程中给予很大的帮助。

Christine 想要感谢她的学生，Paul Bohall，给她介绍了树莓派和她的丈夫 Timothy 以及介绍那些追求“极客思想”的学生给她。

前言

树莓派于 2012 年 2 月一经官方首发就在全球引起了一阵旋风，10000 套设备瞬间售罄。它是一个廉价的只有信用卡大小的裸露电路板，同时，它是一个运行开源 Linux 操作系统的完全可编程的 PC 系统。树莓派可以连接到互联网上，可以插到电视上，并且它仅需 35 美元。

树莓派最初只是为了激发学龄儿童对计算机的兴趣，但是它在世界范围内引起了极客、企业家和教育家的广泛关注。

树莓派的官方编程语言是 Python。Python 是一种灵活的编程语言，可以运行在任何平台上。因此，程序可以在 Windows PC 或者 Mac 上创建并运行在树莓派上，反之亦然。Python 是一种优雅、可靠、功能强大、而且非常流行的编程语言。使用 Python 作为树莓派的官方编程语言是一个非常正确的决定。

用 Python 编程

本书的目标是帮助并引导学生和爱好者在树莓派使用 Python 编程语言。你不需要任何的编程经验就能从本书中获益，我们会完成所有必要的步骤来让你创建自己的 Python 程序并且让它运行起来。

第一部分，“树莓派编程环境”将会详细介绍树莓派核心系统以及如何使用已经安装好的 Python 环境。第 1 章“配置树莓派”展示了如何配置树莓派系统，然后在第 2 章“认识 Raspbian Linux 发行版”中，我们会仔细看看专门为树莓派设计的 Linux 发行版 Raspbian。第 3 章“搭建编程环境”将会详细讲解使用不同的方法在树莓派上运行你的 Python 程序，并且它会提供一些关于如何构建你的程序的小技巧。

第二部分，“Python 基础知识”主要关注 Python v3 编程语言。Python v3 是 Python 的最新版，并且它被树莓派完全支持。从第 4 章到第 7 章将带你学习 Python 编程的基础知识，从简单的赋值语句（第 4 章“Python 基础”）、算法（第 5 章“在程序中使用算术”），和结构化命令（第 6 章“控制你的程序”），到复杂的结构化命令（第 7 章“循环”）。

第 8 章“使用列表和元组”和第 9 章“字典和集合”揭开了第三部分的序幕，“高级 Python”展示了如何使用 Python 提供的一些奇特的数据结构，如元组、字典以及集合。你将会在你的 Python 程序中大量使用，学会这些东西会很有帮助。

在第 10 章“使用字符串”我们将花一点额外的时间看一下 Python 是如何处理文本字符串的。字符串处理是 Python 的一大特点，因此要让你明白所有的这些是如何工作的。

完成了这些入门内容后，我们将了解 Python 中一些更复杂的概念：使用文件（第 11 章“使用文件”），创建你自己的函数（第 12 章“创建函数”）、创建你自己的模块（第 13 章“使用模块”）、面向对象的 Python 编程（第 14 章“探索面向对象编程的世界”）、继承（第 15 章“使用继承”）、正则表达式（第 16 章“正则表达式”）、以及处理异常（第 17 章“异常处理”）。

第四部分“图形编程”是关于如何用 Python 创建一个真实世界的应用。第 18 章“GUI（Graphical User Interface, GUI）编程”讨论了如何用 GUI 编程创建你自己的视窗程序。最后，第 19 章，“游戏编程”会为你介绍 Python 游戏编程的世界。

第五部分，在“业务编程”中我们会看一下如何创建一些面向业务的程序。在第 20 章“使用网络”中我们将会看到如何把一些网络功能，如邮件或者从网页上获取数据的功能集成到你的 Python 程序中。第 21 章“在程序中使用数据库”展示了如何跟一些流行的 Linux 数据库服务器交互。第 22 章“Web 编程”演示了如何编写一个能在网上访问的 Python 程序。

第六部分“树莓派 Python 项目”将详细讲解一些针对树莓派的特性的 Python 工程。第 23 章“创建基础的树莓派 Python 工程”展示了如何使用树莓派的视频和声音功能创建一个多媒体工程。第 24 章“树莓派/Python 高级项目”探索如何通过通用输入/输出（General Purpose Input/Output, GPIO）接口将你的树莓派和电子电路连接起来。

谁应该阅读本书

本书主要面向于那些希望通过编写自己的 Python 程序来让树莓派发挥最大作用的读者，可以细分为 3 个群体。

- 希望通过一个低成本的方法学习 Python 编程的学生。
- 想发挥树莓派最大功能的业余爱好者。
- 希望寻找一个用来开发应用的低成本 Linux 平台的企业家。

如果你正在阅读本书，你可能不是一个编程新手，但是你可能不熟悉使用 Python 编程。

本书体例

步骤

在本书中，我们将很多代码分解成容易理解的一步一步的程序。

文件名、文件夹名称以及代码

这些内容都会以 monospace 字体呈现。

命令

命令和它们的语法都使用粗体

菜单命令

对于所有程序菜单命令我们都是用下面的格式：菜单，命令。这里的目录指的是你打开的下拉菜单，命令是你选择的命令。这里有一个例子：文件，打开。这表示你打开文件菜单并选择了打开命令。

同时，在书中会使用以下几种方框提醒你一些重要的或者有意思的信息。

提示

提示会出现在当前主题的旁边来提供给你额外的信息。这些部分内容会提供一些额外的见解、有用的建议、解决问题的方法或是快捷方式。

警告

警告会提醒你那些会带来严重后果的动作或者误操作。

目 录

第一部分 树莓派编程环境

第 1 章 配置树莓派	3
1.1 树莓派是什么	3
1.1.1 树莓派的历史	3
1.1.2 为什么要学习用 Python 在树莓派上进行编程	4
1.2 获取树莓派	4
1.3 哪些树莓派外设是必须的	6
1.3.1 SD 卡	6
1.3.2 电源	7
1.3.3 显示输出	8
1.3.4 键盘	9
1.4 其他不错的外设	9
1.4.1 挑选一个外壳	9
1.4.2 使用 USB 鼠标	10
1.4.3 自供电 USB 集线器	10
1.4.4 使用网线或者无线 适配器	10
1.5 决定如何购买外设	11
1.6 让你的树莓派正常工作	11
1.6.1 自己研究一下	11
1.6.2 选择操作系统	11
1.6.3 下载操作系统	12
1.6.4 将操作系统部署到 SD 卡上	12

1.6.5 连接外设	13
1.7 排除树莓派的故障	14
1.7.1 检查外设连接线	14
1.7.2 检查 SD 卡	15
1.7.3 检查你的操作系统镜像	15
1.7.4 检查你的外设	15
1.8 总结	15
1.9 Q&A	15
1.10 练习	16
1.10.1 问题	16
1.10.2 答案	16
第 2 章 认识 Raspbian Linux 发行版	17
2.1 了解 Linux	17
2.2 使用 Raspbian 命令行	18
2.3 使用 Raspbian 图形用户界面	22
2.3.1 LXDE 图形界面	23
2.3.2 LXPanel	24
2.4 总结	29
2.5 Q&A	29
2.6 练习	29
2.6.1 问题	29
2.6.2 答案	30
第 3 章 搭建编程环境	31
3.1 探索 Python	31
3.1.1 Python 简史	31

3.1.2	Python v3 和 Python v2	32	4.4	Python 变量进行赋值	60
3.2	检查你的 Python 环境	32	4.4.1	格式化变量和字符串 输出	60
3.2.1	检查 Python 解释器和 交互式 shell	32	4.4.2	避免使用未赋值的变量	61
3.2.2	检查 Python 开发环境	33	4.4.3	将长字符串赋值给变量	61
3.2.3	检查文本编辑器	33	4.4.4	给变量赋予更多类型 的值	62
3.3	安装 Python 和工具	34	4.4.5	给变量重新赋值	62
3.4	关于 Python 解释器	35	4.5	关于 Python 数据类型	63
3.5	关于 Python 交互式 shell	36	4.6	获取用户输入	64
3.6	关于 Python 开发环境 shell	38	4.7	总结	69
3.7	创建 Python 脚本	42	4.8	Q&A	69
3.7.1	在交互式 shell 里运行 Python 脚本	42	4.9	练习	69
3.7.2	使用 IDLE 创建一个 脚本	44	4.9.1	问题	69
3.7.3	使用文本编辑器创建一个 Python 脚本	45	4.9.2	答案	69
3.8	使用适当的工具	46	第 5 章	在程序中使用算术	70
3.9	总结	47	5.1	使用数学运算符	70
3.10	Q&A	47	5.1.1	Python 数学操作符	70
3.11	练习	48	5.1.2	运算符的优先级	72
3.11.1	问题	48	5.1.3	在数学运算中使用变量	73
3.11.2	答案	48	5.1.4	浮点数的精度	74
			5.1.5	显示数字	74
			5.1.6	运算符缩写	75
			5.2	使用分数	75
			5.2.1	分数对象	75
			5.2.2	分数操作	76
			5.3	使用复数	76
			5.3.1	创建实数	77
			5.3.2	复数运算	77
			5.4	math 模块的高级功能	77
			5.4.1	数值函数	78
			5.4.2	幂和对数函数	78
			5.4.3	三角函数	79
			5.4.4	双曲函数	80
			5.4.5	统计数学函数	80
			5.5	使用 NumPy 数学库	80
			5.5.1	NumPy 数据类型	81
			5.5.2	创建 NumPy 数组	81
			5.5.3	使用 NumPy 数组	82
			5.6	总结	82
第二部分 Python 基础知识					
第 4 章	Python 基础	51			
4.1	脚本的输出	51			
4.1.1	探索 print 函数	51			
4.1.2	使用字符作为 print 的 参数	52			
4.1.3	格式化 print 函数的 输出	53			
4.1.4	使用转义序列控制输出	54			
4.1.5	好玩的东西	55			
4.2	格式化脚本	56			
4.2.1	长文本行	57			
4.2.2	创建注释	57			
4.3	理解 Python 变量	58			
4.3.1	Python 关键字	59			
4.3.2	创建 Python 变量名	59			

5.7	Q&A	83	7.5	总结	111
5.8	练习	83	7.6	Q&A	112
5.8.1	问题	83	7.7	练习	112
5.8.2	答案	83	7.7.1	问题	112
第 6 章	控制你的程序	84	7.7.2	答案	112
6.1	使用 if 语句	84	第三部分 高级 Python		
6.2	组合多个语句	85	第 8 章	使用列表和元组	115
6.3	通过 else 语句添加其他条件	86	8.1	关于元组	115
6.4	使用 elif 添加更多的条件	88	8.1.1	创建元组	115
6.5	在 Python 中比较值	89	8.1.2	访问元组中的数据	116
6.5.1	数字比较	89	8.1.3	访问一个范围内的值	117
6.5.2	字符串比较	90	8.1.4	使用元组	117
6.5.3	列表和元组比较	91	8.2	关于列表	119
6.5.4	布尔值比较	92	8.2.1	创建一个列表	119
6.5.5	评估函数返回值	92	8.2.2	使用列表解析	120
6.6	检查复杂的条件	93	8.2.3	从列表中获取数据	120
6.6.1	使用逻辑运算符	93	8.2.4	使用列表	120
6.6.2	组合条件检查	94	8.3	使用多维列表存储数据	125
6.7	对条件检查取反	94	8.4	在脚本中使用列表和元组	125
6.8	总结	95	8.4.1	遍历一个列表或元组	125
6.9	Q&A	95	8.4.2	排序和倒序再	126
6.10	练习	95	8.5	使用列表解析创建列表	127
6.10.1	问题	95	8.6	关于 range 类型	127
6.10.2	答案	96	8.7	总结	128
第 7 章	循环	97	8.8	Q&A	128
7.1	执行重复的任务	97	8.9	练习	129
7.2	使用 for 循环进行迭代	97	8.9.1	问题	129
7.2.1	遍历列表中的数字	98	8.9.2	答案	129
7.2.2	注意一些“陷阱”	99	第 9 章	字典和集合	130
7.2.3	由值指定的数据类型	100	9.1	理解 Python 字典	130
7.2.4	遍历字符串列表	101	9.2	字典基础	131
7.2.5	使用变量进行遍历	102	9.2.1	创建一个字典	131
7.2.6	使用 range 函数进行 迭代	102	9.2.2	填充字典	131
7.3	使用 while 循环	106	9.2.3	更新一个字典	134
7.3.1	使用字符串作为判断条件 进行遍历	106	9.2.4	管理一个字典	135
7.3.2	使用 while True	108	9.3	用字典编程	135
7.4	创建嵌套循环	110	9.4	理解 Python 集合	139
			9.5	集合基础	140

9.6 从集合获取信息	141	11.2.1 指定打开模式	166
9.6.1 集合成员	141	11.2.2 使用文件对象方法	167
9.6.2 并集	141	11.3 读取文件	168
9.6.3 交集	142	11.3.1 读取整个文件	168
9.6.4 差集	142	11.3.2 逐行读取文件	169
9.6.5 对称差集	143	11.3.3 不按顺序读取文件	170
9.6.6 遍历集合	143	11.4 关闭一个文件	173
9.7 修改一个集合	143	11.5 写文件	174
9.8 用集合编程	145	11.5.1 创建并且写入一个 新文件	174
9.9 总结	147	11.5.2 写入到已存在的文件	177
9.10 Q&A	147	11.6 总结	178
9.11 练习	148	11.7 Q&A	179
9.11.1 问题	148	11.8 练习	179
9.11.2 答案	148	11.8.1 问题	179
第 10 章 使用字符串	149	11.8.2 答案	179
10.1 字符串的基础知识	149	第 12 章 创建函数	181
10.1.1 字符串格式	149	12.1 在程序中使用 Python 函数	181
10.1.2 创建字符串	150	12.1.1 创建函数	182
10.1.3 处理字符串	151	12.1.2 使用函数	182
10.2 使用函数操作字符串	152	12.2 返回值	184
10.2.1 改变字符串值	152	12.3 给函数传值	185
10.2.2 分割字符串	153	12.3.1 传递参数	185
10.2.3 连接字符串	153	12.3.2 设置默认参数值	187
10.2.4 测试字符串	154	12.3.3 处理可变数量的参数	188
10.2.5 检索字符串	155	12.3.4 使用字典检索值	189
10.3 格式化字符串输出	156	12.4 在函数中处理变量	190
10.3.1 format()函数	157	12.4.1 局部变量	190
10.3.2 格式化数字	158	12.4.2 全局变量	191
10.4 总结	161	12.5 在函数中使用列表	192
10.5 Q&A	161	12.6 递归函数	193
10.6 练习	161	12.7 总结	194
10.6.1 问题	161	12.8 Q&A	194
10.6.2 答案	162	12.9 练习	195
第 11 章 使用文件	163	12.9.1 问题	195
11.1 理解 Linux 文件结构	163	12.9.2 答案	195
11.1.1 看一下 Linux 目录	164	第 13 章 使用模块	196
11.1.2 通过 Python 管理文件 和目录	165	13.1 介绍模块概念	196
11.2 打开文件	166	13.2 探索标准模块	198

13.3	学习 Python 模块	199	15.3.1	创建子类	227
13.4	创建定制的模块	202	15.3.2	将子类添加到对象模块文件中	228
13.4.1	创建或收集函数放到一起	203	15.3.3	添加其他子类	229
13.4.2	确定模块的名称	203	15.3.4	将子类方法放到它自己的对象模块文件中	230
13.4.3	在测试目录中创建这个自定义模块	203	15.4	在 Python 脚本中使用继承	231
13.4.4	测试自定义模块	204	15.5	总结	241
13.4.5	将模块移动到生产目录	205	15.6	Q&A	241
13.4.6	检查路径, 如果需要则修改	207	15.7	练习	241
13.4.7	测试生产级别自定义模块	207	15.7.1	问题	241
13.5	总结	210	15.7.2	答案	242
13.6	Q&A	210	第 16 章	正则表达式	243
13.7	练习	211	16.1	什么是正则表达式	243
13.7.1	问题	211	16.1.1	定义正则表达式	243
13.7.2	答案	211	16.1.2	正则表达式的类型	244
第 14 章	探索面向对象编程的世界	212	16.2	在 Python 中使用正则表达式	244
14.1	理解面向对象编程基础	212	16.2.1	正则表达式函数	244
14.1.1	什么是 OOP	212	16.2.2	编译正则表达式	246
14.1.2	定义一个类	213	16.3	定义基本的模式	246
14.1.3	创建一个实例	213	16.3.1	纯文本	247
14.1.4	默认属性值	214	16.3.2	特殊字符	248
14.2	定义类方法	214	16.3.3	锚字符	248
14.2.1	设值方法	215	16.3.4	点字符	250
14.2.2	访问方法	215	16.3.5	字符分类	250
14.2.3	添加辅助方法	217	16.3.6	字符类取反	252
14.3	使用类模块共享你的代码	220	16.3.7	使用范围	252
14.4	总结	222	16.3.8	星号	253
14.5	Q&A	222	16.4	使用高级正则表达式特性	253
14.6	练习	223	16.4.1	问号	253
14.6.1	问题	223	16.4.2	加号	254
14.6.2	答案	223	16.4.3	使用大括号	254
第 15 章	使用继承	224	16.4.4	管道符号	254
15.1	了解类的问题	224	16.4.5	分组表达式	255
15.2	理解子类和继承	225	16.5	在 Python 脚本中使用正则表达式	255
15.3	在 Python 中使用继承	226	16.6	总结	257
			16.7	Q&A	258
			16.8	练习	258

16.8.1	问题	258
16.8.2	答案	258
第 17 章	异常处理	259
17.1	理解异常	259
17.1.1	语法错误异常	259
17.1.2	运行时错误异常	261
17.2	处理异常	263
17.3	处理多个异常	265
17.3.1	创建多个 try except 语句块	267
17.3.2	处理通用的异常	269
17.3.3	理解 try except 语句的 选项	270
17.4	总结	275
17.5	Q&A	275
17.6	练习	275
17.6.1	问题	275
17.6.2	答案	276
 第四部分 图形编程		
第 18 章	GUI 编程	279
18.1	为 GUI 环境编程	279
18.1.1	视窗接口	280
18.1.2	事件驱动编程	280
18.2	检查 Python 的 GUI 包	281
18.3	使用 tkinter 包	281
18.3.1	创建一个窗口	281
18.3.2	向窗口添加控件	282
18.3.3	定义事件处理 程序	286
18.4	探索 tkinter 控件	288
18.4.1	使用 Label 控件	288
18.4.2	添加 Button 控件	288
18.4.3	使用 Checkbutton 控件	289
18.4.4	使用 Entry 控件	290
18.4.5	添加 Text 控件	292
18.4.6	使用 ListBox 控件	293
18.4.7	使用 Menu 控件	295
18.5	总结	297
18.6	Q&A	298
18.7	练习	298
18.7.1	问题	298
18.7.2	答案	298
第 19 章	游戏编程	299
19.1	理解游戏编程	299
19.2	学习游戏相关的框架和库	300
19.3	配置 PyGame 库	300
19.3.1	检查 PyGame	301
19.3.2	安装 PyGame	301
19.3.3	确保你的系统是 最新的	301
19.3.4	安装构建 Python 模块的 工具	302
19.3.5	获取 PyGame 的源码	303
19.3.6	安装额外的软件包	303
19.3.7	构建和安装 PyGame	304
19.3.8	使用 PyGame	304
19.3.9	加载和初始化 PyGame	306
19.3.10	配置游戏屏幕	307
19.3.11	向游戏画面中放置 文本	308
19.4	学习 PyGame 的更多内容	311
19.4.1	一直在游戏中	311
19.4.2	绘制图像和形状	311
19.4.3	在游戏中使用声音	314
19.5	处理 PyGame 动作	315
19.5.1	在游戏画面中移动 图形	315
19.5.2	与游戏画面中的图形 交互	316
19.6	总结	322
19.7	Q&A	323
19.8	练习	323
19.8.1	问题	323
19.8.2	答案	323

第五部分 业务编程

第 20 章 使用网络	327
20.1 寻找 Python 的网络模块	327
20.2 与邮件服务器一起工作	328
20.2.1 Linux 世界中的邮件	328
20.2.2 smtplib 库	330
20.2.3 使用 smtplib 库	330
20.3 与网页服务器一起工作	334
20.3.1 获取网页	334
20.3.2 解析网页数据	335
20.4 使用套接字编程连接应用 程序	338
20.4.1 什么是套接字编程	338
20.4.2 Python 的 socket 模块	339
20.4.3 创建服务器程序	340
20.4.4 创建客户端程序	342
20.4.5 运行客户端/服务器 示例	343
20.5 总结	344
20.6 Q&A	344
20.7 练习	344
20.7.1 问题	344
20.7.2 答案	344
第 21 章 在程序中使用数据库	345
21.1 使用 MySQL 数据库	345
21.1.1 安装 MySQL	345
21.1.2 配置 MySQL 环境	346
21.1.3 安装 Python 的 MySQL 模块	350
21.1.4 创建 Python 脚本	351
21.2 使用 PostgreSQL 数据库	354
21.2.1 安装 PostgreSQL	354
21.2.2 配置 PostgreSQL 环境	354
21.2.3 安装 Python 的 PostgreSQL 模块	357
21.2.4 用 pycpg2 编写 代码	358
21.3 总结	360

21.4 Q&A	360
21.5 练习	360
21.5.1 问题	360
21.5.2 答案	361
第 22 章 Web 编程	362
22.1 在树莓派上运行 Web 服务器	362
22.1.1 安装 Apache Web 服务器	363
22.1.2 提供 HTML 文件	364
22.2 公网网关接口编程	365
22.2.1 什么是 CGI	365
22.2.2 运行 Python 程序	365
22.3 扩展 Python 网页	367
22.3.1 格式化输出	367
22.3.2 使用动态网页	368
22.3.3 调试 Python 程序	370
22.4 处理表单	372
22.4.1 创建网页表单	373
22.4.2 cgi 模块	374
22.5 总结	376
22.6 Q&A	376
22.7 练习	377
22.7.1 问题	377
22.7.2 答案	377

第六部分 树莓派 Python 项目

第 23 章 创建基础的树莓派 Python 工程	381
23.1 思考基础的树莓派 Python 工程	381
23.2 通过 Python 显示高清图片	381
23.2.1 理解高分辨率	382
23.2.2 使用脚本演示图像	383
23.2.3 配置演示画面	384
23.2.4 查找图片	384
23.2.5 在可移动的驱动器上 存储照片	385
23.2.6 缩放照片	388

23.2.7	照片取景	389
23.2.8	居中照片	389
23.2.9	提高演示速度	391
23.2.10	潜在的脚本修改	396
23.3	播放音乐	397
23.3.1	创建基本的音乐脚本	397
23.3.2	将音乐存储在可移动的硬盘上	398
23.3.3	使用音乐播放列表	398
23.3.4	控制回放	399
23.3.5	让播放列表随机播放	403
23.4	创建一个特殊的演示	403
23.5	总结	407
23.6	Q&A	407
23.7	练习	407
23.7.1	问题	407
23.7.2	答案	407
第 24 章	树莓派/Python 高级项目	408
24.1	探索 GPIO 接口	408
24.1.1	什么是 GPIO 接口	408
24.1.2	GPIO 引脚布局	409
24.1.3	连接 GPIO	410
24.2	使用 RPi.GPIO 模块	412
24.2.1	安装 RPi.GPIO	412
24.2.2	启动方法	413
24.3	控制 GPIO 输出	413

24.3.1	配置硬件来查看 GPIO 输出	414
24.3.2	测试 GPIO 输出	415
24.3.3	闪烁 LED	416
24.3.4	创建一个花式的闪光灯	417
24.4	检测 GPIO 输入	417
24.4.1	配置检测输入的硬件	418
24.4.2	处理输入信号	419
24.4.3	输入轮询	420
24.4.4	输入事件	421
24.5	总结	423
24.6	Q&A	423
24.7	练习	423
24.7.1	问题	423
24.7.2	答案	423

第七部分 附录

附录	将 Raspbian 操作系统加载到 SD 卡上	427
Windows:	将 Raspbian 加载到 SD 卡上	427
Linux:	将 Raspbian 加载到 SD 卡上到 SD 卡上	429
Mac:	将 Raspbian 加载到 SD 卡上	431