

开发有趣的极客项目，掌握实用的Python编程技能

异步图书  
www.epubit.com.cn



# Python极客项目编程

## PYTHON PLAYGROUND

[美] Mahesh Venkitachalam 著 王海鹏 译



中国工信出版集团

人民邮电出版社  
POSTS & TELECOM PRESS



# Python 极客项目编程

## PYTHON PLAYGROUND

[美] Mahesh Venkitachalam 著 王海鹏 译



人民邮电出版社  
北京

## 图书在版编目 ( C I P ) 数据

Python极客项目编程 / (美) 温科特卡姆  
(Mahesh Venkitachalam) 著 ; 王海鹏译. -- 北京 : 人  
民邮电出版社, 2017. 5  
ISBN 978-7-115-44976-4

I. ①P… II. ①温… ②王… 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043822号

## 版 权 声 明

Simplified Chinese-language edition copyright © 2017 by Posts and Telecom Press.

Copyright © 2016 by Mahesh Venkitachalam. Title of English-language original: Python Playground , ISBN-13:  
978-1-59327-604-1, published by No Starch Press.

All rights reserved.

本书中文简体字版由美国 No Starch 出版社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。

- 
- ◆ 著 [美] Mahesh Venkitachalam
  - 译 王海鹏
  - 责任编辑 陈冀康
  - 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: 19.5  
字数: 441 千字 2017 年 5 月第 1 版  
印数: 1-3 000 册 2017 年 5 月北京第 1 次印刷  
著作权合同登记号 图字: 01-2015-8784 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

广告经营许可证: 京东工商广字第 8052 号

## 内容提要

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，我们能够解决现实生活中的很多任务。

本书通过 14 个有趣的项目，帮助和鼓励读者探索 Python 编程的世界。全书共 14 章，分别介绍了通过 Python 编程实现的一些有趣项目，包括解析 iTunes 播放列表、模拟人工生命、创建 ASCII 码艺术图、照片拼接、生成三维立体图、创建粒子模拟的烟花喷泉效果、实现立体光线投射算法，以及用 Python 结合 Arduino 和树莓派等硬件的电子项目。本书并不介绍 Python 语言的基础知识，而是通过一系列不简单的项目，展示如何用 Python 来解决各种实际问题，以及如何使用一些流行的 Python 库。

本书适合那些想要通过 Python 编程来进行尝试和探索的读者，适合了解基本的 Python 语法和基本的编程概念的读者进一步学习，对于 Python 程序员有一定的启发和参考价值。

## 致谢

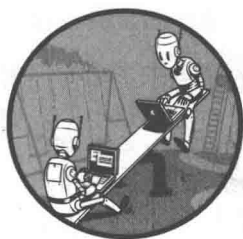
写一本书就像跑马拉松。有人这么对我说过。但我确实知道，写这本书考验了我的耐力极限，没有亲朋好友和家人为我摇旗呐喊，我不可能完成。

首先，我感谢我的妻子 Hema，她不变的爱、鼓励和耐心，贯穿了完成这项工作的整整两年时间。我感谢我的朋友 Raviprakash Jayaraman，他是我所有不确定的项目的“同谋”，也是本书的技术评审，我们一起吃了许多有趣的午餐、看了不少电影，多次去逛 S.P. Road Zoo。我感谢我的朋友 Seby Kallarakkal，他推动我编写这本书，进行了多次有趣的讨论。我很感谢我的朋友 Santosh Hemachandra 博士在快速傅里叶变换上的有益讨论。我感谢 Karthikeyan Chellappa，帮助我测试 Python 模块的安装，和我一起围绕 Kaikondrahalli 湖跑步。我还要感谢 Matthew Denham（我与他在 Reddit 上交谈），他对万花尺（Spirograph）的数学知识提供了帮助。

我感谢 No Starch 出版社的 Tyler Ortman 和 Bill Pollock，他们接受了我编写本书的意向。还要感谢 Serena Yang 编辑本书的专业工作。我感谢 Nicholas Kramer 对本书的技术评阅。

我感谢我的父母 A.V. Venkitachalam 和 N. Saraswathy，他们为我提供的教育远远超越了他们的财务能力。最后，我感谢所有给我启发的老师，我希望一辈子做一名学生。

# 前 言



欢迎阅读本书!在本书中,你会看到 14 个令人兴奋的项目,旨在鼓励你探索 Python 编程的世界。这些项目涉及广泛的主题,如绘制类似万花尺的花纹、生成 ASCII 码艺术图、3D 渲染,以及根据音乐同步投射激光图像。除了本身很有趣之外,这些项目的意图是提供一些起点,让你通过扩展每个项目,来探索你自己的想法。

## 本书的目标读者

本书的目标读者,是所有想知道如何利用编程来理解和探索想法的人。本书的项目假设你了解基本的 Python 语法和基本的编程概念,并假设你熟悉高中数学知识。我已经尽了最大的努力,详细解释了所有项目中需要的数学知识。

本书不会是你的第一本 Python 书。我不会指导你学习基本知识。但我会通过一系列不简单的项目,向你展示如何用 Python 来解决各种实际问题。在学习这些项目时,你将探索 Python 编程语言的细微差别,并学习如何使用一些流行的 Python 库。但也许更重要的是,你将学习如何将问题分解成几个部分,开发一个算法来解决这个问题,然后从头用 Python 来实现一个解决方案。解决现实世界的问题可能很难,因为它们往往是开放式的,并且需要各个领域的专业知识。但 Python 提供了一些工具,协助解决问题。克服困难,寻找实际问题的解决方案,这是成为专家级程序员的旅途中最重要的环节。

## 本书的内容

让我们快速浏览一下本书各章的内容。

## 第一部分：热身运动

第 1 章展示了如何解析 iTunes 播放列表文件，并从中收集有用的信息，如音轨长度和共同的音轨。在第 2 章中，我们使用参数方程及海龟作图法，绘制类似万花尺产生的那些曲线。

## 第二部分：模拟生命

这部分是用数学模型来模拟现象。在第 3 章中，我们将学习如何实现 Conway 游戏的生命游戏算法，产生动态的模式来创建其他模式，以模拟一种人工生命。第 4 章展示了如何用 Karplus-Strong 算法来创建逼真的弹拨音。然后，在第 5 章中，我们将学习如何实现类鸟群算法，模拟鸟类的聚集行为。

## 第三部分：图像之乐

这部分介绍使用 Python 读取和操作 2D 图像。第 6 章展示了如何根据图像创建 ASCII 码艺术图。在第 7 章中，我们将进行照片拼接。在第 8 章中，我们将学习如何生成三维立体图，它让人产生 3D 图像的错觉。

## 第四部分：走进三维

这一部分的项目使用 OpenGL 的 3D 图形库。第 9 章介绍使用 OpenGL 创建简单 3D 图形的基本知识。在第 10 章中，我们将创建粒子模拟的烟花喷泉，它用数学和 OpenGL 着色器来计算和渲染。在第 11 章中，我们将使用 OpenGL 着色器来实现立体光线投射算法，来渲染立体数据，该技术常用于医疗影像，如 MRI 和 CT 扫描。

## 第五部分：玩转硬件

在最后一部分中，我们将用 Python 来探索 Arduino 微控制器和树莓派。在第 12 章中，我们将利用 Arduino，通过一个简单电路读取并标绘传感器数据。在第 13 章中，我们将利用 Python 和 Arduino 来控制两个旋转镜和激光器，生成响应声音的激光秀。在第 14 章中，我们将使用树莓派打造一个基于网络的气象监测系统。

## 为何选择 Python

Python 是探索编程的理想语言。作为一种多范式语言，在如何组织程序方面，它提供了极大的灵活性。你可以将 Python 视为脚本语言，简单地执行代码，或将其视为过程语言，把程序组织成一组彼此调用的函数，或将其视为面向对象语言，利用类、继承和模块来建立层次结构。这种灵活性让你可以选择最适合特定项目的编程风格。

如果用更传统的语言来开发，如 C 或 C ++，你必须先编译和链接代码，然后才能运行它。使用 Python，你可以编辑后直接运行它（在背后，Python 将你的代码编译成中间字节码，然后由 Python 解释器运行，但这些过程对用户是透明的）。在实践中，用 Python 多次修改并运行代码，要容易很多。

此外，Python 解释器是非常方便的工具，可用于检查代码语法，获得模块的帮助，进行快速计算，甚至测试在开发中的代码。例如，我写 Python 代码时，会打开三个窗口：文本编辑器、命令行和 Python 解释器。我在编辑器中写代码时，会在解释器中导入我的函数或类，边开发边测试。

Python 有一组非常小、简单而强大的数据结构。如果你理解了字符串、列表、元组、字典、列表解析和基本控制结构，如 for 和 while 循环，那么你已经开了个好头。Python 简洁而有表现力的语法，使得我们很容易只用几行代码，就完成复杂的操作。而一旦熟悉 Python 内置的模块和第三方模块，你将拥有大量的工具，用于解决真正的问题，就像本书中介绍的那样。从 Python 中调用 C/C++ 代码有标准的方式，反之亦然。因为在 Python 中可以找到库来做几乎所有事情，我们很容易在大型项目中组合使用 Python 和其他语言模块。这就是为什么 Python 被认为是了不起的胶水语言，它可以很容易地组合使用不同的软件组件。本书最后的硬件项目展示了 Python 如何与 Arduino 和 JavaScript 代码协作。真实的软件项目经常使用多种软件技术，Python 非常适合这种分层体系结构。

下面的例子展示了 Python 的易用性。在第 14 章中为树莓派天气监控器开发代码时，我看着温度/湿度传感器的示波器输出，写下这一串数字：

---

```
001101110000000000011010000000001010001
```

---

因为我不能用二进制讲话，所以启动了 Python 解释器并输入：

---

```
>>> str = '001101110000000000011010000000001010001'
>>> len(str)
40
>>> [int(str[i:i+8], 2) for i in range(0, 40, 8)]
[55, 0, 26, 0, 81]
```

---

这行代码将 40 位字符串切分转换成 5 个 8 位的整数，这是我可以理解的。上述数据被解释为 55.0% 的湿度，温度为 26.0 摄氏度，校验和是  $55 + 26 = 81$ 。

这个例子展示了如何实际使用 Python 解释器作为非常强大的计算器。你不必写一个完整的程序就能快速计算，只要打开解释器，就可以开始。这只是我喜欢 Python 的一个原因，原因还有很多，所以我认为你也会喜欢 Python。

## Python 的版本

本书基于 Python 3.3.3，但所有代码都与 Python 2.7.x 和 3.x 兼容。



## 本书的代码

在本书中，我尽了最大的努力引导你详细研究每个项目的代码，一段接一段地进行。你可以自己输入代码，或从 <https://github.com/electronut/pp/> 或 [www.epubit.com.cn](http://www.epubit.com.cn) 下载书中所有程序的完整源代码（单击 Download Zip 选项）。

接下来，你会看到一些令人兴奋的项目。我希望你玩这些项目时，享受的乐趣和我创造它们时一样多。而且不要忘记，利用每个项目结束时提供的练习进一步探索。我希望你和本书一起度过许多快乐的编程时光！

# 目 录

## 第一部分 热身运动

第 1 章 解析 iTunes 播放列表.....	3	2.1.2 海龟画图.....	19
1.1 iTunes 播放列表文件剖析.....	3	2.2 所需模块.....	20
1.2 所需模块.....	5	2.3 代码.....	20
1.3 代码.....	5	2.3.1 Spiro 构造函数.....	20
1.3.1 查找重复.....	5	2.3.2 设置函数.....	21
1.3.2 提取重复.....	6	2.3.3 restart()方法.....	21
1.3.3 查找多个播放列表中 共同的音轨.....	7	2.3.4 draw()方法.....	22
1.3.4 收集统计信息.....	8	2.3.5 创建动画.....	22
1.3.5 绘制数据.....	8	2.3.6 SpiroAnimator 类.....	23
1.3.6 命令行选项.....	9	2.3.7 genRandomParams()方法.....	24
1.4 完整代码.....	10	2.3.8 重新启动程序.....	24
1.5 运行程序.....	13	2.3.9 update()方法.....	25
1.6 小结.....	14	2.3.10 显示或隐藏光标.....	25
1.7 实验.....	14	2.3.11 保存曲线.....	25
第 2 章 万花尺.....	15	2.3.12 解析命令行参数和初始化.....	26
2.1 参数方程.....	16	2.4 完整代码.....	27
2.1.1 万花尺方程.....	17	2.5 运行万花尺动画.....	32
		2.6 小结.....	33
		2.7 实验.....	33

## 第二部分 模拟生命

第3章 Conway 生命游戏	37
3.1 工作原理	38
3.2 所需模块	39
3.3 代码	40
3.3.1 表示网格	40
3.3.2 初始条件	41
3.3.3 边界条件	41
3.3.4 实现规则	42
3.3.5 向程序发送命令行参数	43
3.3.6 初始化模拟	43
3.4 完整代码	44
3.5 运行模拟人生的游戏	46
3.6 小结	47
3.7 实验	47
第4章 用 Karplus-Strong 算法产生 音乐泛音	49
4.1 工作原理	51
4.1.1 模拟	51
4.1.2 创建 WAV 文件	52
4.1.3 小调五声音阶	53
4.2 所需模块	54
4.3 代码	54
4.3.1 用 deque 实现环形缓冲区	54
4.3.2 实现 Karplus-Strong 算法	55

4.3.3 写 WAV 文件	56
4.3.4 用 pygame 播放 WAV 文件	56
4.3.5 main()方法	57
4.4 完整代码	58
4.5 运行拨弦模拟	61
4.6 小结	62
4.7 实验	62
第5章 类鸟群: 仿真鸟群	63
5.1 工作原理	64
5.2 所需模块	64
5.3 代码	64
5.3.1 计算类鸟群的位置和速度	65
5.3.2 设置边界条件	66
5.3.3 绘制类鸟群	67
5.3.4 应用类鸟群规则	68
5.3.5 添加个体	70
5.3.6 驱散类鸟群	71
5.3.7 命令行参数	71
5.3.8 Boids 类	71
5.4 完整代码	72
5.5 运行类鸟群模拟	75
5.6 小结	76
5.7 实验	76

## 第三部分 图片之乐

第6章 ASCII 文本图形	79
6.1 工作原理	80
6.2 所需模块	81
6.3 代码	81
6.3.1 定义灰度等级和网格	82

6.3.2 计算平均亮度	82
6.3.3 从图像生成 ASCII 内容	83
6.3.4 命令行选项	84
6.3.5 将 ASCII 文本图形字符 串写入文本文件	84

6.4	完整代码	85	7.3.8	控制照片马赛克的大小	97
6.5	运行 ASCII 文本图形生成程序	87	7.4	完整代码	98
6.6	小结	87	7.5	运行照片马赛克生成程序	102
6.7	实验	88	7.6	小结	103
<b>第 7 章</b>	<b>照片马赛克</b>	<b>89</b>	7.7	实验	103
7.1	工作原理	90	<b>第 8 章</b>	<b>三维立体画</b>	<b>105</b>
7.1.1	分割目标图像	90	8.1	工作原理	106
7.1.2	平均颜色值	91	8.1.1	感知三维立体画中的深度	106
7.1.3	匹配图像	91	8.1.2	深度图	108
7.2	所需模块	92	8.2	所需模块	109
7.3	代码	92	8.3	代码	109
7.3.1	读入小块图像	92	8.3.1	重复给定的平铺图像	109
7.3.2	计算输入图像的平均 颜色值	93	8.3.2	从随机圆创建平铺图像	110
7.3.3	将目标图像分割成网格	93	8.3.3	创建三维立体画	111
7.3.4	寻找小块的最佳匹配	94	8.3.4	命令行选项	112
7.3.5	创建图像网格	95	8.4	完整代码	113
7.3.6	创建照片马赛克	96	8.5	运行三维立体画生成程序	115
7.3.7	添加命令行选项	97	8.6	小结	117
			8.7	实验	117

## 第四部分 走进三维

<b>第 9 章</b>	<b>理解 OpenGL</b>	<b>121</b>	9.4.3	Scene 类	133
9.1	老式 OpenGL	122	9.5	完整代码	137
9.2	现代 OpenGL: 三维图形管线	124	9.6	运行 OpenGL 应用程序	142
9.2.1	几何图元	124	9.7	小结	143
9.2.2	三维变换	125	9.8	实验	143
9.2.3	着色器	127	<b>第 10 章</b>	<b>粒子系统</b>	<b>145</b>
9.2.4	顶点缓冲区	128	10.1	工作原理	146
9.2.5	纹理贴图	129	10.1.1	为粒子运动建模	147
9.2.6	显示 OpenGL	129	10.1.2	设置最大范围	147
9.3	所需模块	130	10.1.3	渲染粒子	149
9.4	代码	130	10.1.4	利用 OpenGL 混合来 创建更逼真火花	149
9.4.1	创建 OpenGL 窗口	130	10.1.5	使用公告板	150
9.4.2	设置回调	131			

10.1.6	生成火花动画	151	11.3	项目代码概述	178
10.2	所需模块	151	11.4	生成三维纹理	178
10.3	粒子系统的代码	151	11.5	完整的三维纹理代码	180
10.3.1	定义粒子的几何形状	152	11.6	生成光线	181
10.3.2	为粒子定义时间延迟 数组	153	11.6.1	定义颜色立方体的 几何形状	182
10.3.3	设置粒子初始速度	153	11.6.2	创建帧缓冲区对象	184
10.3.4	创建顶点着色器	154	11.6.3	渲染立方体的背面	185
10.3.5	创建片段着色器	156	11.6.4	渲染立方体的正面	185
10.3.6	渲染	156	11.6.5	渲染整个立方体	186
10.3.7	Camera 类	158	11.6.6	调整大小处理程序	187
10.4	粒子系统完整代码	158	11.7	完整的光线生成代码	187
10.5	盒子代码	164	11.8	体光线投射	192
10.6	主程序代码	166	11.8.1	顶点着色器	194
10.6.1	每步更新这些粒子	167	11.8.2	片段着色器	194
10.6.2	键盘处理程序	168	11.9	完整的体光线投射代码	196
10.6.3	管理主程序循环	168	11.10	二维切片	199
10.7	完整主程序代码	169	11.10.1	顶点着色器	201
10.8	运行程序	172	11.10.2	片段着色器	202
10.9	小结	172	11.10.3	针对二维切片的 用户界面	202
10.10	实验	172	11.11	完整的二维切片代码	203
第 11 章	体渲染	173	11.12	代码整合	206
11.1	工作原理	174	11.13	完整的主文件代码	207
11.1.1	数据格式	174	11.14	运行程序	209
11.1.2	生成光线	175	11.15	小结	210
11.1.3	显示 OpenGL 窗口	177	11.16	实验	210
11.2	所需模块	178			

## 第五部分 玩转硬件

第 12 章	Arduino 简介	215	12.2.3	社区	218
12.1	Arduino	216	12.2.4	外设	219
12.2	Arduino 生态系统	217	12.3	所需模块	219
12.2.1	语言	218	12.4	搭建感光电路	219
12.2.2	IDE	218	12.4.1	电路工作原理	219

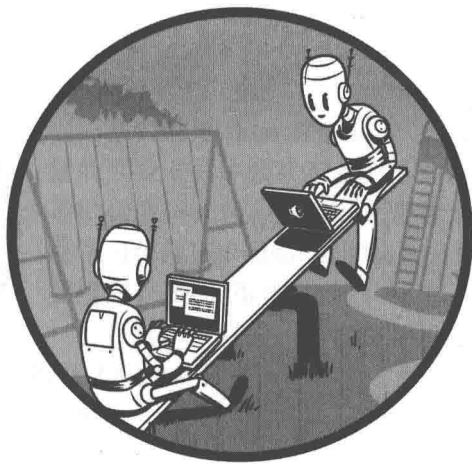
12.4.2	Arduino 程序	220	13.7	小结	250
12.4.3	创建实时图表	221	13.8	实验	250
12.5	Python 代码	222	<b>第 14 章 基于树莓派的天气监控器</b> 253		
12.6	完整的 Python 代码	224	14.1	硬件	254
12.7	运行程序	226	14.1.1	DHT11 温湿度传感器	254
12.8	小结	227	14.1.2	树莓派	255
12.9	实验	227	14.1.3	设置树莓派	255
<b>第 13 章</b>	<b>激光音乐秀</b>	229	14.2	安装和配置软件	256
13.1	用激光产生图案	230	14.2.1	操作系统	257
13.1.1	电机控制	230	14.2.2	初始配置	257
13.1.2	快速傅里叶变换	232	14.2.3	Wi-Fi 设置	257
13.2	所需模块	233	14.2.4	设置编程环境	258
13.2.1	搭建激光秀	234	14.2.5	通过 SSH 连接	259
13.2.2	连接电机驱动器	236	14.2.6	Web 框架 Bottle	259
13.3	Arduino 程序	237	14.2.7	用 flot 绘制	260
13.3.1	配置 Arduino 数字 输出引脚	238	14.2.8	关闭树莓派	261
13.3.2	主循环	238	14.3	搭建硬件	262
13.3.3	停止电机	240	14.4	代码	263
13.4	Python 代码	240	14.4.1	处理传感器数据请求	264
13.4.1	选择音频设备	241	14.4.2	绘制数据	264
13.4.2	从输入设备读取数据	241	14.4.3	update()方法	267
13.4.3	计算数据流的 FFT	242	14.4.4	用于 LED 的 JavaScript 处理程序	267
13.4.4	从 FFT 值提取频率 信息	243	14.4.5	添加交互性	268
13.4.5	将频率转换为电机 速度和方向	243	14.5	完整代码	269
13.4.6	测试电机设置	244	14.6	运行程序	272
13.4.7	命令行选项	245	14.7	小结	273
13.4.8	手动测试	245	14.8	实验	273
13.5	完整的 Python 代码	246	<b>附录 A 软件安装</b> 275		
13.6	运行程序	249	<b>附录 B 基础实用电子学</b> 281		
			<b>附录 C 树莓派的建议和技巧</b> 289		

# 第一部分

## 热身运动

“在初学者的头脑中有很多可能性，  
在专家的头脑中，可能性很少。”

——铃木俊隆

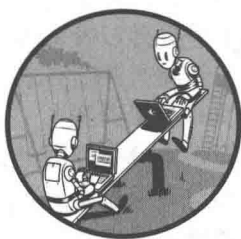






# 第 1 章

## 解析 iTunes 播放列表



我们的 Python 探险始于一个简单的项目，该项目在 iTunes 播放列表文件中查找重复的乐曲音轨，并绘制各种统计数据，如音轨长度和评分。你可以从查看 iTunes 播放列表格式开始，然后学习如何用 Python 提取这些文件的信息。为了绘制这些数据，要用到 matplotlib 库。

在这个项目中，我们将学习以下主题：

- XML 和属性列表（p-list）文件；
- Python 列表和字典；
- 使用 Python 的 set 对象；
- 使用 numpy 数组；
- 直方图和散点图；
- 用 matplotlib 库绘制简单的图；
- 创建和保存数据文件。

### 1.1 iTunes 播放列表文件剖析

iTunes 资料库中的信息可以导出为播放列表文件（在 iTunes 中选择 File ▶ Library ▶