

图书基本信息

书名：《趣学Python编程》

13位ISBN编号：9787115335951

10位ISBN编号：7115335958

出版时间：2014-3

出版社：人民邮电出版社

作者：Jason Briggs

页数：268

译者：尹哲

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

《趣学Python编程》

内容概要

python是一款解释型、面向对象、动态数据类型的高级程序设计语言。python语法简捷而清晰，具有丰富和强大的类库，因而在各种行业中得到广泛的应用。对于初学者来讲，python是一款既容易学又相当有用的编程语言，国内外很多大学开设这款语言课程，将python作为一门编程语言学习。

《趣学python编程》是一本轻松、快速掌握python编程的入门读物。全书分为3部分，共18章。第1部分是第1章到第12章，介绍python编程基础知识，包括python的安装和配置、变量、字符串、列表、元组和字典、条件语句、循环语句函数和模块、类、内建函数和绘图，等等。第2部分是第13章和第14章，介绍如何用python开发实例游戏弹球。第3部分包括第15章到第18章，介绍了火柴人实例游戏的开发过程。

《趣学python编程》语言轻松，通俗易懂，讲解由浅入深，力求将读者阅读和学习的难度降到最低。任何对计算机编程有兴趣的人或者首次接触编程的人，不论孩子还是成人，都可以通过阅读本书来学习python编程。

作者简介

jason r. briggs，从8岁开始编写程序，他学习的第一种编程语言是在radio shack trs-80上的basic。作为开发人员及系统架构师，他是职业软件开发者，同时他又是《java开发者》杂志的特约编辑。他的文章上过javaworld、onjava以及onlamp。这是他撰写的第一本书。

书籍目录

第1部分 学习编程

第1章 python不是大蟒蛇 3

1.1 关于计算机语言 3

1.2 安装python 4

1.2.1 在windows 7上安装python 4

1.2.2 在苹果os x上安装python 6

1.2.3 在ubuntu上安装python 8

1.3 当你安装好python以后 9

1.4 保存python程序 10

1.5 你学到了什么 12

第2章 计算与变量 13

2.1 用python来做计算 13

2.1.1 python的运算符 14

2.1.2 运算的顺序 15

2.2 变量就像是标签 16

2.3 使用变量 17

2.4 你学到了什么 20

第3章 字符串、列表、元组和字典 21

3.1 字符串 21

3.1.1 创建字符串 21

3.1.2 处理字符串相关的问题 23

3.1.3 在字符串里嵌入值 25

3.1.4 字符串乘法 26

3.2 列表比字符串还强大 27

3.2.1 添加元素到列表 29

3.2.2 从列表中删除元素 30

3.2.3 列表上的算术 31

3.3 元组 32

3.4 python里的map不是用来指路的 33

3.5 你学到了什么 35

3.6 编程小测验 36

第4章 用海龟画图 37

4.1 使用python的turtle (海龟) 模块 37

4.1.1 创建画布 38

4.1.2 移动海龟 39

4.2 你学到了什么 43

4.3 编程小测验 43

第5章 用if和else来提问 45

5.1 if语句 45

5.2 语句块就是一组程序语句 45

5.3 条件语句帮助我们做比较 48

5.4 if-then-else语句 49

5.5 if和elif语句 50

5.6 组合条件 51

5.7 没有值的变量——none 52

5.8 字符串与数字之间的不同 52

5.9 你学到了什么 55

5.10 编程小测验	55
第6章 循环	57
6.1 使用for循环	57
6.2 还有一种叫while的循环	64
6.3 你学到了什么	67
6.4 编程小测验	67
第7章 使用函数和模块来重用你的代码	69
7.1 使用函数	69
7.1.1 函数的组成部分	70
7.1.2 变量和作用域	71
7.2 使用模块	73
7.3 你学到了什么	75
7.4 编程小测验	76
第8章 如何使用类和对象	78
8.1 把事物拆分成类	78
8.1.1 父母与孩子	79
8.1.2 增加属于类的对象	80
8.1.3 定义类中的函数	81
8.1.4 用函数来表示类的特征	81
8.1.5 为什么使要用类和对象	82
8.1.6 画图的对象与类	84
8.2 对象和类的另一些实用功能	86
8.2.1 函数继承	87
8.2.2 从函数里调用其他函数	88
8.3 初始化对象	89
8.4 你学到了什么	90
8.5 编程小测验	90
第9章 python的内建函数	92
9.1 使用内建函数	92
9.1.1 abs函数	92
9.1.2 bool函数	93
9.1.3 dir函数	94
9.1.4 eval函数	96
9.1.5 exec函数	97
9.1.6 float函数	98
9.1.7 int函数	98
9.1.8 len函数	99
9.1.9 max和min函数	100
9.1.10 range函数	101
9.1.11 sum函数	102
9.2 使用文件	103
9.2.1 创建测试文件	103
9.2.2 在python中打开文件	106
9.2.3 写入到文件	107
9.3 你学到了什么	108
9.4 编程小测验	108
第10章 常用的python模块	109
10.1 使用copy模块来复制	109
10.2 keyword模块记录了所有的关键字	112

- 10.3 用random模块获得随机数 112
 - 10.3.1 用randint来随机挑选一个数字 112
 - 10.3.2 用choice从列表中随机选取一个元素 114
 - 10.3.3 用shuffle来给列表洗牌 115
- 10.4 用sys模块来控制shell程序 115
 - 10.4.1 用exit函数来退出shell程序 115
 - 10.4.2 从stdin对象读取 115
 - 10.4.3 用stdout对象来写入 116
 - 10.4.4 我用的python是什么版本的 116
- 10.5 用time模块来得到时间 117
 - 10.5.1 用asctime来转换日期 118
 - 10.5.2 用localtime来得到日期和时间 119
 - 10.5.3 用sleep来休息一会儿吧 120
- 10.6 用pickle模块来保存信息 120
- 10.7 你学到了什么 122
- 10.8 编程小测验 122
- 第11章 高级海龟作图 124
 - 11.1 从基本的正方形开始 124
 - 11.2 画星星 125
 - 11.3 画汽车 128
 - 11.4 填色 130
 - 11.4.1 用来画填色圆形的函数 131
 - 11.4.2 使用纯白和纯黑 132
 - 11.5 画方形的函数 133
 - 11.6 画填色正方形 134
 - 11.7 画填好色的星星 135
 - 11.8 你学到了什么 137
 - 11.9 编程小测验 137
- 第12章 用tkinter画高级图形 139
 - 12.1 创建一个可以点的按钮 140
 - 12.2 使用具名参数 142
 - 12.3 创建一个画图用的画布 142
 - 12.4 画线 143
 - 12.5 画盒子 144
 - 12.5.1 画许多矩形 146
 - 12.5.2 设置颜色 148
 - 12.6 画圆弧 151
 - 12.7 画多边形 153
 - 12.8 显示文字 154
 - 12.9 显示图片 155
 - 12.10 创建基本的动画 157
 - 12.11 让对象对操作有反应 159
 - 12.12 更多使用id的方法 161
 - 12.13 你学到了什么 163
 - 12.14 编程小测验 163
- 第2部分 弹球实例
- 第13章 你的第一个游戏：弹球 167
 - 13.1 击打反弹球 167
 - 13.2 创建游戏的画布 168

- 13.3 创建ball类 169
- 13.4 增加几个动作 171
 - 13.4.1 让小球移动 171
 - 13.4.2 让小球来回反弹 172
 - 13.4.3 改变小球的起始方向 174
- 13.5 你学到了什么 176
- 第14章 完成你的第一个游戏：反弹吧，小球！ 177
 - 14.1 加上球拍 177
 - 14.2 让球拍移动 179
 - 判断小球是否击中球拍 180
 - 14.3 增加输赢因素 183
 - 14.4 你学到了什么 187
 - 14.5 编程小测验 187
- 第3部分 火柴人实例
- 第15章 火柴小人游戏的图形 191
 - 15.1 火柴小人游戏计划 191
 - 15.2 得到gimp 192
 - 15.3 创建游戏中的元素 193
 - 15.3.1 准备一个有透明背景的图形 194
 - 15.3.2 画火柴人 195
 - 15.3.3 画平台 197
 - 15.3.4 画门 197
 - 15.3.5 画背景 198
 - 15.3.6 透明 199
 - 15.4 你学到了什么 200
- 第16章 开发火柴人游戏 201
 - 16.1 创建game类 201
 - 16.1.1 设置窗口标题以及创建画布 201
 - 16.1.2 完成__init__函数 202
 - 16.1.3 创建主循环函数 203
 - 16.2 创建坐标类 205
 - 16.3 冲突检测 205
 - 16.3.1 精灵在水平方向上冲突 206
 - 16.3.2 精灵在垂直方向上冲突 208
 - 16.3.3 把它们放在一起：最终的冲突检测代码 208
 - 16.4 创建精灵类 210
 - 16.5 添加平台类 211
 - 16.5.1 加入平台对象 212
 - 16.5.2 添加很多平台 213
 - 16.6 你学到了什么 215
 - 16.7 编程小测验 215
- 第17章 创建火柴人 217
 - 17.1 初始化火柴人 217
 - 17.1.1 装入火柴人图形 217
 - 17.1.2 设置变量 218
 - 17.1.3 与键盘按键绑定 219
 - 17.2 让火柴人向左转和向右转 220
 - 17.3 让火柴人跳跃 221
 - 17.4 我们都做了什么 221

17.5 你学到了什么	222
第18章 完成火柴人逃生游戏	223
18.1 让火柴人动起来	223
18.1.1 创建动画函数	223
18.1.2 得到火柴人的位置	226
18.1.3 让火柴人移动	227
18.2 测试我们的火柴人精灵	235
18.3 门	235
18.3.1 创建doorsprite类	236
18.3.2 门的检测	237
18.3.3 加入门对象	237
18.4 最终的游戏	238
18.5 你学到了什么	244
18.6 编程小测验	245
结束语 接下来学什么	246
附录 python的关键字	253
术语表	265

精彩短评

- 1、英文名python for kids，这是一本简单的python书籍，后面还有两个简单的游戏 不错适合第一次看，这本看完了 下面开始看A BYTE OF PYTHON
- 2、挺有意思，零基础学习，非常适合我。照着写了个弹球游戏，很开心。不过Python for Kids？什么样的kids能看懂啊。。。
- 3、适合零基础
- 4、真的是对小孩子写的一本书，内容浅显，覆盖面小。
- 5、入门级很不错，还能编小游戏玩
- 6、适合青少年的入门书
- 7、非常入门
- 8、跟着写代码还是很舒服
- 9、For Kids...整本书都萌萌哒
- 10、后面两个游戏(paddleball, 火柴人)的例子很好
- 11、啊啊啊，掩面而泣
- 12、好书。有趣。
- 13、比较适合高中 大一或者更小孩子的入门书吧 为了让孩子理解有些东西说得 你懂得。。。用游戏启发兴趣也不错 不过我最后几章扫一眼就过去了 总之适合完全没有编程基础的人看吧
- 14、Python3比Python2要难，不适合未成年人阅读，家长们慎买
- 15、书写得不及格，毫无特点
- 16、兴趣入门还行，想学好还是老老实实都官方文档吧
- 17、希望以后，某个风雨交加的夜晚，闲来无事，我教孩子写Python，多美好
- 18、浅显易懂，最后做了两个小游戏，但对于学习语言来说，知识点还是太少了，可以做趣味阅读和兴趣入门
- 19、2016.12.28.孩子，我们一起“玩”电脑吧。
- 20、迅速上手利器，好用。英文名却叫“Python for kids”，掩面读完。
- 21、作为一本初学Python的入门书来说，马马虎虎。适合编程经验非常少的初学者看。简单看完只有一个感受：当初第一门语言学Python该多好.....
- 22、我觉得挺好的
- 23、在读
- 24、其实看着挺好玩的，简单、有趣。等把手上的3本都读完再横向比较下。
- 25、有用，已看到第五章了，感觉不难呢

精彩书评

- 1、作为一名互联网老兵，能够深刻感受到计算机和互联网的普及所带来的天翻地覆的变化和冲击。程序员用代码正在改变着所有的行业。未来，指挥计算机去完成你想做的事情，就像指挥自己的腿如何走路一样重要！我有一个8岁的儿子，从小就喜欢玩电脑，今年我告诉他，所有的电脑游戏都是程序员写出来的，从而激发了他对程序的兴趣，央求着要学编程。为此，我找了很多编程书籍，大多数理论知识太多，所以他很难理解，至于儿子有不想继续学下去，直到朋友推荐的这一本（python 趣学忘记名字了），直接从一个游戏实例来入手，很容易让小孩了解程序通过代码可以让电脑画图或者让图片动起来，就像游戏里场景一样，然后再引申出原理和语法规则，经过几天的学习儿子完全入迷，3个月就学完了整本教材，而且能够活学活用根据课程内容作出自己的小游戏。这是一本好的教材，希望能出系列版本，能够引导孩子完全掌握python。——转自网友亲身体会
- 2、原书是好书，特别是原作者对“类”和“对象”的讲解，简单生动，初学者一看就明白。但是.....举个例子告诉你本书的翻译质量：“让我们创建一个函数来显示在每一周到一年内我们可以压平多少罐子。”多读几遍，能明白么？实际上原文的意思是：“创建一个函数显示一年的52周，每周可压得罐子数”。"Let ' s create a function to show how many cans we ' ve flattenedeach week up to a year."再比如：“"游戏玩家要提供写成python小程序的对机器人的指令。”照汉语的习惯一般是这样说：“玩家会提供一小段python代码作为操作机器人的指令集。”“Players of the game would provide the instructions for their robot as mini Python programs.”第九章习题，译者不知道every other是什么意思本书翻译：“分别打印出下面字符串的每个单词”。英文原文是“print every other word in the following string”，意思应该是“每隔一词打印出下面的字符串”。真让十几岁的孩子学这种不负责任的翻译，肯定一头雾水了。类似的情况不胜枚举。总而言之，原书很好，本中文版并不适合给孩子读。

章节试读

1、《趣学Python编程》的笔记-第1页

<https://www.nostarch.com/pythonforkids>

```
turtle画星星import turtle  
t = turtle.Pen()
```

```
def mystar(size, filled):  
    if filled:  
        t.begin_fill()  
    for x in range(1,19):  
        t.forward(size)  
        if x % 2 == 0:  
            t.left(175)  
        else:  
            t.left(225)  
    if filled:  
        t.end_fill()
```

```
t.color(0.9, 0.75, 0)  
mystar(120, True)  
t.color(0,0,0)  
mystar(120, False)
```

```
turtle基本动画示例import time  
from tkinter import *
```

```
tk = Tk()  
canvas = Canvas(tk, width=400, height=200)  
canvas.pack()  
canvas.create_polygon(10, 10, 10, 60, 50, 35)  
for x in range(0, 60):  
    canvas.move(1, 5, 0)  
    tk.update()  
    time.sleep(0.05)
```

```
弹球from tkinter import *  
import random  
import time
```

```
tk = Tk()  
tk.title("Game")  
tk.resizable(0, 0)  
tk.wm_attributes("-topmost", 1)  
canvas = Canvas(tk, width=500, height=400, bd=0, highlightthickness=0)  
canvas.pack()  
tk.update()
```

class Ball:

```
def __init__(self, canvas, paddle, color):
    self.canvas = canvas
    self.paddle = paddle
    self.id = canvas.create_oval(10, 10, 25, 25, fill=color)
    self.canvas.move(self.id, 245, 100)
    starts = [-3, -2, -1, 1, 2, 3]
    random.shuffle(starts)
    self.x = starts[0]
    self.y = -3
    self.canvas_height = self.canvas.winfo_height()
    self.canvas_width = self.canvas.winfo_width()
    self.hit_bottom = False
```

def draw(self):

```
self.canvas.move(self.id, self.x, self.y)
pos = self.canvas.coords(self.id)
if pos[1] <= 0:
    self.y = 3
if self.hit_paddle(pos) == True:
    self.y = -3
if pos[3] >= self.canvas_height:
    self.hit_bottom = True
if pos[0] <= 0:
    self.x = 3
if pos[2] >= self.canvas_width:
    self.x = -3
```

def hit_paddle(self, pos):

```
paddle_pos = self.canvas.coords(self.paddle.id)
if pos[2] >= paddle_pos[0] and pos[0] <= paddle_pos[2]:
    if pos[3] >= paddle_pos[1] and pos[3] <= paddle_pos[3]:
        return True
    return False
```

class Paddle:

```
def __init__(self, canvas, color):
    self.canvas = canvas
    self.id = canvas.create_rectangle(0, 0, 100, 10, fill=color)
    self.canvas.move(self.id, 200, 300)
    self.x = 0
    self.canvas_width = self.canvas.winfo_width()
    self.canvas.bind_all('<KeyPress-Left>', self.turn_left)
    self.canvas.bind_all('<KeyPress-Right>', self.turn_right)
```

def turn_left(self, evt):

```
self.x = -2
```

def turn_right(self, evt):

```
self.x = 2
```

```
def draw(self):  
    self.canvas.move(self.id, self.x, 0)  
    pos = self.canvas.coords(self.id)  
    if pos[0] <= 0:  
        self.x = 0  
    elif pos[2] >= self.canvas_width:  
        self.x = 0
```

```
paddle = Paddle(canvas, 'blue')
```

```
ball = Ball(canvas, paddle, 'red')
```

```
while 1:
```

```
    if ball.hit_bottom == False:
```

```
        ball.draw()
```

```
        paddle.draw()
```

```
    tk.update_idletasks()
```

```
    tk.update()
```

```
    time.sleep(0.01)
```

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:www.tushu000.com