

STEAM & 创客教育趣学指南

达人迷®

Python
FOR
KIDS

Python

趣味编程10例

轻松学习新技能：

- 学习 Python 编程语言
- 创作游戏
- 定制属于自己的项目

◎ [美] Brendan Scott 著

◎ 东洋 译

◎ 程晨 审



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

for dummies
A Wiley Brand

STEAM & 创客教育趣学指南



◎ [美] Brendan Scott 著

◎ 东洋 译

◎ 程晨 审

for
dummies
A Wiley Brand

人 民 邮 电 出 版 社

北 京

图书在版编目(CIP)数据

达人迷. Python趣味编程10例 / (美) 布伦丹·斯科特 (Brendan Scott) 著; 东洋译. — 北京: 人民邮电出版社, 2018. 6

(STEAM&创客教育趣学指南)

ISBN 978-7-115-47680-7

I. ①达… II. ①布… ②东… III. ①软件工具—程序设计 IV. ①TP31

中国版本图书馆CIP数据核字(2018)第000440号

版权声明

Original English language edition Copyright©2015 by Wiley Publishing, Inc. All rights reserved including the right of reproduction in whole or in part in any form. This translation published by arrangement with Wiley Publishing, Inc.

本书原英文版本版权©2015 归 Wiley Publishing, Inc.所有。未经许可不得以任何形式全部或部分复制作品。本书中文简体字版是经过与 Wiley Publishing, Inc.协商出版。

商标声明

Wiley, the Wiley Publishing Logo, For Dummies, the Dummies Man and related trade dress are trademarks or registered trademarks of John Wiley and Sons, Inc. and/or its affiliates in the United States and/or other countries. Used under license. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

Wiley、Wiley Publishing 徽标、For Dummies、the Dummies Man 以及相关的商业特殊标志均为 John Wiley and Sons, Inc.及/或其子公司在美国和/或其他国家的商标或注册商标, 未经许可不得使用所有其他商标均为其各自所有者的财产。本书中提及的任何产品或供应商与 John Wiley & Sons, Inc.及出版社无关。

◆ 著 [美] Brendan Scott

译 东 洋

责任编辑 周 璇

责任印制 周昇亮

◆ 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>

北京捷迅佳彩印刷有限公司印刷

◆ 开本: 800×1000 1/16

印张: 14.75

2018 年 6 月第 1 版

字数: 249 千字

2018 年 6 月北京第 1 次印刷

著作权合同登记号 图字: 01-2016-2468 号

定价: 89.00 元

读者服务热线: (010)81055339 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广登字 20170147 号

内容提要

本书以生动诙谐的语言，图文并茂地讲解了 Python 的核心入门知识，从 Python 软件如何安装、配置开始，再到测试第一个简单的程序“Hello World”，然后进阶到游戏设计环节，最终实现几个高难度的完整项目的制作。初学者一步一步跟着学，就可以通过实践掌握 Python 的应用技术。本书适合对编程感兴趣的青少年和大众初学者阅读。

作者简介

Brendan 是 3 个孩子的父亲。他非常爱他们，他也非常爱 Python，因此他开通博客来教他的大孩子学习如何编程。这可是在 2010 年啊！这件事使得这个网站火了，现在他有机会把对 Python 的喜爱之情传递给更多的孩子们。他真心希望这本书能帮助你掌握 Python。

本书英文版出版致谢

执行编辑: Steven Hayes

发展编辑: Tonya Maddox Cupp

技术编辑: Camille McCue

高级编辑助理: Cherie Case

项目协调员: Antony Sami

特别技术帮助: Dexter Lim, Lewis Lim

封面图片: ©Wiley

目 录

概述	1
关于本书	1
格式说明	1
给达人迷们的假设	3
本书使用的图标	3
本书之外的信息	4
马上开始	4
第1周 了解 Python	5
第1章 初识 Python	7
Python 是什么？它为什么这么神奇？	8
谁在使用 Python？	9
使用 Python	10
理解本书的教学方法	11
面面俱到	11
带你编程带你飞	12
注重实用	12
在 Mac OS X 系统上安装 Python	13
在 Windows 系统上安装 Python	13
使用 Firefox 浏览器	13
使用 IE 浏览器	14
在 Linux 操作系统上安装 Python	16
把 Python 固定到开始菜单	16
启动 Python 解释器	18
使用 Python 内置文档	19
终止 Python 解释器运行	19
查找 Python 在线文档	20
Python 在线文档	20
Python 内省特性	21
Professor 网络	21

源代码	22
加入更广阔的 Python 社区	22
PEPs	22
Planet Python 与 PyCon	23
处理问题	23
如何去学习	24
实践	24
犯错	24
思考	25
Ganbatte Ne !	25
总结	25
第 2 章 构建第一个 Python 程序: Hello World!	26
编写 Hello World!	26
定位和修复错误	27
使用文字常量	29
使用变量保存字符串常量	30
中断程序执行	33
Python 的关键字	36
许多循环, 许多 Hello	36
用问候填满屏幕	37
让 Python 计数	38
总结	40
第 2 周 构建猜谜游戏	41
第 3 章 构建一个猜谜游戏	43
设计游戏	43
从选手那里获取输入	44
要求输入	44
确保变量是相等的	45
调用运算符	46
深入 Python	48
将猜测与数字比较	49
对比选手的猜测和计算机想出的数字	51
告知选手猜测是否是错误的	52
一直尝试, 直到选手猜对	54

让 Python 产生随机数	56
使用命名空间	58
完成猜谜游戏	59
总结	60
第 4 章 设置编程环境	61
使用默认的开发环境	61
启动 IDLE	62
一些 IDLE 的小技巧	64
tab 补全	64
命令历史记录	65
使用 IDLE 的编辑窗口	67
在文件中写注释	69
插入哈希注释	70
保存命令行内容	71
注释代码	71
缩进和取消缩进	73
总结	74
第 5 章 构建一个更完美的猜谜游戏	75
操作你的函数	75
给函数命名	78
为函数增加帮助文档	78
为函数坐桩	80
重构猜谜游戏	81
查找逻辑问题	83
解决逻辑问题	84
注意 computers_number 的二次应用	85
理解作用域的原理	86
与函数通信	89
给函数发送消息	89
赋予参数默认值	91
以函数通信	92
添加分数	94
让用户退出	95
完整的代码	98
总结	100

第3周 创造文字游戏	101
第6章 黑客对讲机: 1337 Sp34k3r	103
字符串中有个对象	103
引用对象的属性	106
了解列表	107
遍历列表中的所有元素	107
创建自己的列表	109
创建列表	110
验证元素是否在列表中	111
设计自己的精英黑客对讲机	112
创建文件	112
创建函数	113
运行代码	113
替换	114
替换一个字母	115
让用户输入消息	116
定义字符替换	117
替换所有字符串	119
使用 print 调试代码	120
使用 IDLE 的调试器	123
总结	126
第7章 加密	128
删掉那些没用的字符	129
制作一张替换表	130
创建你自己的加密器	131
使用字典	133
创建一个加密字典	134
使用 join	136
重写加密函数	137
编写解密函数	138
创建一个解密字典	139
双向测试	140
输入原文本或者加密文本	143
加密一个文本文件	144
打开、写入并关闭一个文件	145

读取文件	146
运行文件	147
使用 with 语法打开或者关闭	147
从文件中加密解密	148
选择一个名称并创建测试数据	149
打开文件加密数据	149
在 Shell 中解密	151
更改解密部分的代码	153
完整的代码	155
总结	156
第 8 章 无厘头的句子	158
插入格式化字符串	158
格式化值的个数要满足要求	159
使用元组数据类型	160
开始构造无厘头的句子	162
填充模板	164
添加更多的单词	166
完整的代码	167
总结	168
第 4 周 学习面向对象编程	169
第 9 章 地址簿	171
第一个类对象	172
创建一个类	172
创建一个实例	173
创建类和实例属性	174
规划地址簿	175
创建文件并添加一个类	175
添加第一条信息	177
使用 __init__ 实例化一个类	178
创建一个打印实例的函数	180
使用魔术方法 __repr__	181
初始化 AddressBook 实例	183
发现 pickle 的力量	185
添加一个 save() 函数	187

在同一个应用程序中载入一个已保存的 pickle	188
创建一个 Controller 类	188
测试 load 方法	191
添加接口	193
编写方法的具体内容	196
完整代码	198
总结	202
第 10 章 算术训练器	203
制定制作算数训练器的计划	203
伊始	204
创建问题	206
按行提问	208
将问题随机排列	208
每次提出指定个数的问题	210
打印乘法表	212
跨屏幕打印多个表	214
从用户界面说起	216
添加退出功能	218
完善	220
计时	220
整理主循环和其余的部分	222
总结	222

概 述

大家好，欢迎阅读本书，我们的 Python 学习之旅马上开始。如果你能和我一起编写所有的项目，那么阅读完本书，你就掌握了 Python 编程的基础了。

学习本书内容最重要的就是实践。练习编写所有的代码，最好做到在看我写的代码之前，思考出来代码应该是什么样子的。

关于本书

本书将会帮助你了解 Python 编程需要掌握的知识。这里面有很多示例，我也会讲解程序设计的思路。另外我还会帮助你融入更广大的 Python 社区。这样你在学习完本书之后就可以进入 Python 社区了。

格式说明

阅读本书时需要铭记以下几点：

✔ 有时候单词使用斜体字，我会解释斜体字的内容，举个例子：列表中的对象被称为元素。当你阅读到这样的内容时，那你可要仔细观察了，这是概念定义（列表中的对象被称为元素）。

✔ Python 代码的书写字体与其他文字不一样。有时代码会和文字夹杂在一起，例如：`print('Hello World!')`。

✔ 有时候代码会单独写成一段，例如：

```
print('Hello World!')
```

✔ 有些代码行以 `>>>` 开始。这是我在展示使用交互式 Python 解析器时出现的提示符。

你需要在自己的 Python 解析器内输入书上 `>>>` 提示符之后的代码，然后验证程序执行结果：

```
>>> my_message = "Hello World!"
>>> print(my_message)
```

✓ 每行代码之前的空格数非常重要。代码的长度（理论上来说）无关紧要，但是 Python 的编码规范建议每行代码不要超过 79 个字符（包括字母、数字、空格或者标点符号）。本书可没有你的显示器大，所以每行代码只能写 69 个字符。书里的一些代码被我分成了几行来写。这样代码既能正确执行，看起来也比较美观。所以你在输入的时候要当心，分成几行来写的代码的空格数看起来并不是那么直观。

✓ 我分割代码有两个原则：

1. 第一个原则是隐式的，一般来说，可以在括号内的任意逗号位置分割代码，Python 仍然会把这几行代码认为是一行。被分割开的代码之后的部分应该缩进到和括号对齐的位置，下面以第 9 章中的一段代码为例：

```
values = (e.first_name, e.family_name,
          e.date_of_birth, e.email_address)
```

尽管你将这段代码输入为两行，但 Python 仍然认为这是一行代码（认为这是一行很长的代码）。按照示例输入这段代码。在每行结束的位置输入回车，在每行开始的位置输入空格，对齐到正确的位置。

2. 第二个原则是用反斜线“\”（注意不是 /）明确地分割一行代码。下面是从第 9 章里截取的一个示例：

```
raw_input_prompt = "Press: 1 for training, "+\
                    " 2 for testing, 3 to quit.\n"
```

代码输入成了两行，在第一行的结尾用 \ 分割，不过 Python 仍会将这看作一行。

✓ 当使用 Python 解释器的时候，每行代码以 ... 或者 >>> 开头。如果你在代码中没有看到这些字符，说明之前的代码和当前的代码是同一段代码。下面是项目 2 中的一个例子：

```
>>> my_second_message = 'This name is a little long.
    Ideally, try to keep the name short, but not too
    short.'
```

这段代码的第二行和第三行都没有以 ... 或者 >>> 开头。这意味着只有在输入完所有代码后才能按回车键。也就是说在输入完 too short.' 之后按回车键，而不是在输入 little long. 和 not too 后按回车键。

✓ 有时候，程序在你的计算机上的输出结果与书上的代码看起来略有差异。例如在后面的章节里，可能会遇到重启行。在我的屏幕上，下列代码显示为一行：

```
>>> ===== RESTART
=====
```

- ✔ 第 4 章中会展示如何自动缩进代码。在此之前，每次都需要手动缩进代码。每写一行代码，就需要按 4 次空格键。如果代码需要缩进两层，则需要按 8 次空格键（也就是两层缩进，每层缩进 4 个空格），依此类推。每一行需要缩进的代码都要如此。
- ✔ 展示代码如何执行时，我通常会提供一个代码模板——代码使用概述。一个简单的示例就是 `help([object name])`。这个例子里 `help` 是关键字，它需要和一对小括号配合使用。方括号表明这是可选项。斜体字是需要输入的内容。不是斜体的内容需要照原样输入。按照这个模板，代码 `help(help)` 可以执行（获取关于关键字 `help` 的帮助文档）。代码 `help()` 也是可以执行的，虽然括号里面没有任何内容（因为 `[object name]` 是可选内容）。
- ✔ 网址（URL）和程序代码的字体是 monofont。如果你是在联网的设备上阅读本书，单击地址即可访问那个网站，试试访问 dummies 网站。
- ✔ 有时需要选择菜单栏中的某一项，这个菜单可不是饭店的菜单。例如我让你选择 `File → New File`，意思是从 `File` 菜单里选择 `New File` 选项。
- ✔ `Ctrl` 表示键盘上的 `Ctrl` 键，`Ctrl+A` 组合键的意思是在按住 `Ctrl` 键的同时按住字符 `A` 键，然后同时松开。如果使用 Mac 计算机，键盘上有个控制键，使用这个控制键。`Ctrl-A` 组合键的意思是按住控制键再按字符 `A` 键，然后同时松开两个键。不要使用选择键或者命令键。
- ✔ 如果你使用的是 Mac 计算机，我说的 `Enter` 键就是 Mac 计算机键盘上的 `Return` 键。

给达人迷们的假设

在本书里，我并没有对读者做过多的假设。阅读本书，需要读者至少要能打开计算机，会访问开始菜单（Windows 操作系统）。要想安装 Python 程序，你需要有这台计算机的管理员权限。万事开头难，通读本书需要你下定决心，坚持下去。

本书使用的图标



警告图标提示你要小心！这里面的信息有可能会解决你的大问题。如果不留意这些警告信息的话，可能会丢失数据。



在很长的一段时间里你都要用到此信息，所以要牢记它。



这个图标提示了方便编写程序的快捷信息。



这个图标表明所提示的信息不仅适用于 Python，还广泛适用于编程领域。

本书之外的信息

除了本书还可以找到更多的信息：

- ✓ **备忘单**：本书在网上有一份备忘单，访问 dummies 网站上的 pythonforkids 页面，里面罗列了 Python 的关键词、常见的内置函数以及标准库里选择出的一些函数。当你编写程序时可以作为参考资料。
- ✓ **dummies 网站在线文章和附加项目**：除了本书中的项目之外，网上还有一些附加项目，你可以访问 dummies 网站上的 pythonforkids 页面获取它们。
- ✓ **brendanscott 博客**：访问我的博客。本书许多项目的灵感都起源于那里。每个项目都有自己的博客专题以及其他一些可尝试的东西。如果有什么反馈，可以在博客留言。

马上开始

现在我们马上开始第 1 章，去了解 Python 这个语言能做什么，怎么安装它。在你继续第 2 章之前，确保你已经学会了 Ctrl+C，这样你就准备好写自己的第一个程序了。因为每个章节都是独立的，所以你可以按照自己的喜好在任意项目中穿梭。不过要注意，虽然每个章节不会使用之前项目的代码，但还是会使用到之前项目介绍的概念。

了解 Python