

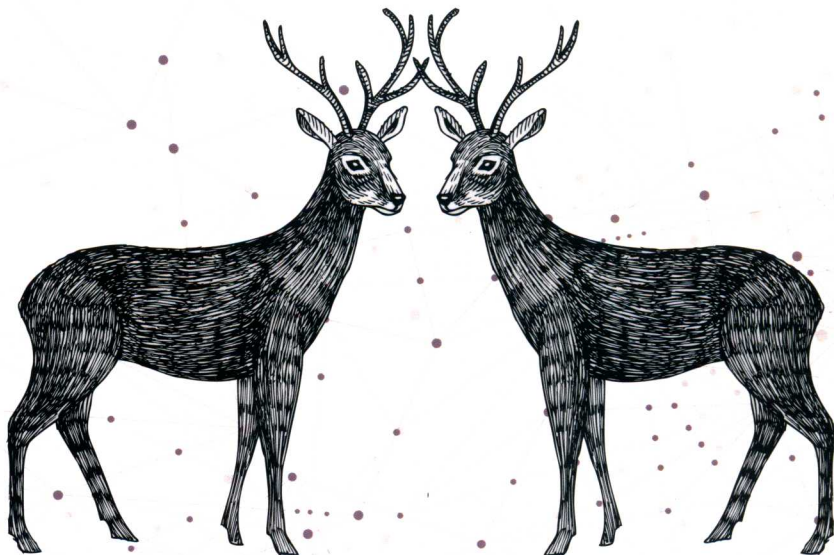
从零基础到具备核心编程技能，通过实战让编程变得更简单

Broadview[®]
www.broadview.com.cn

Python核心编程

从入门到开发实战

朱红庆◎著



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

Python核心编程

从入门到开发实战

朱红庆 著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书以学会、用好 Python 语言进行软件编程为目标，不仅讲解了基本概念、数据类型、变量、运算符、函数、程序的控制结构等 Python 语言的基础知识，还深入介绍了 Python 语言常用库、数据结构、网络编程、可视化编程及图像处理等 Python 语言的核心运用，深入浅出地讲解了 Python 语言的各项技术及实战技能。

全书共 16 章。首先讲解 Python 语言的基本概念、运算符与表达式、变量与数据类型、程序的控制结构等；然后讲解函数、组合数据类型、文件与文件系统、正则表达式、程序进程和线程、Python 操作数据库、Web 网站编程技术、Python 可视化编程；接着重点讲解基于 PyQt 5 技术的 GUI 编程等；最后详细讲解 Python 在交互式游戏开发、智能机器人、人工智能及数据可视化 4 个方面的综合应用。全书不仅融入了作者丰富的工作经验和多年的使用心得，还提供了大量来自工作现场的实例，具有较强的实战性和可操作性。

本书适合那些希望学习 Python 语言编程的初、中级程序员和希望精通 Python 语言编程的高级程序员阅读。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

图书在版编目 (CIP) 数据

Python 核心编程从入门到开发实战 / 朱红庆著. —北京：电子工业出版社，2020.1

ISBN 978-7-121-35705-3

I. ①P… II. ①朱… 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 271147 号

责任编辑：刘 伟 特约编辑：田学清

印 刷：三河市双峰印刷装订有限公司

装 订：三河市双峰印刷装订有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编：100036

开 本：787×980 1/16 印张：18.75 字数：378 千字

版 次：2020 年 1 月第 1 版

印 次：2020 年 1 月第 1 次印刷

定 价：79.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010) 88254888, 88258888。

质量投诉请发邮件至 zlt@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：(010) 51260888-819, faq@phei.com.cn。

前 言

本书针对“零基础”和“入门”级读者，用实例引导读者深入学习，满足其在基础入门、扩展学习、职业技能、项目实战 4 个方面的需求。本书深入浅出地讲解使用 Python 语言进行软件编程中的各项技术及实战技能。读者通过系统学习，可以掌握 Python 语言的基础知识，同时拥有全面的开发能力、优良的团队协作技能和丰富的项目实战经验。

本书内容

全书共 16 章。首先讲解 Python 语言的基本概念、运算符与表达式、变量与数据类型、程序的控制结构等；然后讲解函数、组合数据类型、文件与文件系统、正则表达式、程序进程和线程、Python 操作数据库、Web 网站编程技术、Python 可视化编程；接着重点讲解基于 PyQt 5 技术的 GUI 编程等；最后详细讲解 Python 在交互式游戏开发、智能机器人、人工智能及数据可视化 4 个方面的综合应用。学完本书后，读者将对 Python 在项目开发中的实际应用拥有切身的体会，为日后进行软件开发积累项目管理及实战开发经验。

全书不仅融入了作者丰富的工作经验和多年的使用心得，还提供了大量来自工作现场的实例，具有较强的实战性和可操作性。我们的目标就是让初学者、应届毕业生快速成长为一名合格的初级程序员，通过演练积累项目开发经验和团队合作技能，在未来的职场中站在高的起点，并能迅速融入软件开发团队中。

本书特色

- 结构科学，自学更易。

本书在内容组织和范例设计中都充分考虑到初学者的特点，由浅入深，循序渐进。无论读者是否接触过 Python 语言，都能从本书中找到合适的起点。

- 视频讲解，细致透彻。

为了降低学习难度，提高学习效率，本书录制了同步微视频(模拟培训班模式)，读者通过视频学习，在轻松学会专业知识之余，还能获得软件开发经验，使学习变得更轻松、有趣。

- 超多、实用、专业的范例和实战项目。

本书结合实际工作中的范例，逐一讲解 Python 语言的各种知识，使读者在实战中掌握知识，轻松拥有项目经验。

超值助学资源

本书配备了超值的助学资源库，具体内容如下。

- 助学资源 1：随赠“本书配套学习”资源库，提高读者学习 Python 语言的效率。

- (1) 全书配有 180 多节同步教学微视频，总时长近 18 个学时。
- (2) 全书 4 个大型项目案例及 200 多个范例源代码。
- (3) 书中部分内容有 PPT 电子课件和上机实训教案。

- 助学资源 2：随赠“职业成长”资源库，突破读者职业规划与发展瓶颈。

- (1) 程序员职业规划手册，软件工程师技能手册。
- (2) 面试(笔试)资源库，包括 200 道求职常见面试(笔试)真题与解析。
- (3) 常见错误及解决方案，开发经验及技巧大汇总。
- (4) 200 套求职简历模板，200 套竞聘模板，200 套毕业答辩 PPT 模板。

- 助学资源 3：随赠“Python 语言学习”资源库，拓展学习本书的深度和广度。

- (1) 软件开发模块资源库。
- (2) 项目开发资源库。
- (3) 编程水平测试系统。
- (4) 100 套 Python 典型范例库，40 套项目案例库。

(5) 电子书资源库，包括《Python 关键字速查手册》《Python 标准库查询手册》《Python 常见函数查询手册》《Python 疑难问题速查手册》，以及《Python 语法速查

手册》等。

- 助学资源 4: 随赠在线课程, 可免费学习 Python、Java、JavaScript、C++、Oracle、iOS、Android 等 20 多类 500 余学时的软件开发在线课程。

读者对象

- 没有任何 Python 语言基础的初学者。
- 有一定的 Python 语言基础, 想精通 Python 编程的人员。
- 有一定的 Python 编程基础, 没有项目经验的人员。
- 正在进行毕业设计的学生。
- 大专院校及培训学校的老师和学生。

本书由朱红庆著, 同时参与编写和资料收集的还有朱性强、王闪闪、陈梦、李良、刘静如、陈献凯, 在此一并表示感谢。

在本书的编写过程中, 虽然作者尽可能地将最好的讲解呈现给读者, 但是难免有疏漏和不妥之处, 敬请批评指正。若读者在学习过程中遇到困难或有好的建议, 可发邮件至 elesite@163.com。

说明: 书中部分省略了大家熟知的“语言”二字, 如 Python、C、Java 等, 在本书中分别代表 Python 语言、C 语言和 Java 语言等。

【读者服务】

微信扫码获取



加入本书读者交流群, 与作者交流互动

- 180 多节近 18 个学时的全书同步教学微视频
- 200 多个范例源代码及 4 个大型项目案例源码
- 同步 PPT 电子教学课件及 Python 上机实训案例库
- 100 套 Python 典型范例库, 40 套项目案例库
- 200 道 Python 程序员面试真题解析, 200 套求职简历模板
- 赠送《Python 关键字速查手册》《Python 标准库查询手册》《Python 常见函数查询手册》《Python 疑难问题速查手册》, 以及《Python 语法速查手册》电子书资源库
- 随赠在线课程, 可免费学习 Python、Java、Android 等 20 多类在线课程

目 录

第 1 章 认识 Python 语言	1
1.1 走进 Python.....	1
1.1.1 Python 的优缺点	2
1.1.2 Python 的应用领域.....	4
1.2 建立 Python 程序开发环境.....	4
1.2.1 安装 Python	4
1.2.2 运行 Python 程序	8
1.3 Python 解释器与安装 PyCharm.....	11
1.3.1 Python 解释器.....	12
1.3.2 安装 PyCharm.....	13
第 2 章 快速使用 Python 编程	15
2.1 编程基础知识.....	15
2.2 Python 基本语法元素分析.....	16
2.2.1 程序的层次结构.....	17
2.2.2 代码注释.....	18
2.2.3 代码换行与并行.....	19
2.2.4 变量与保留字	21
2.2.5 赋值语句.....	22
2.2.6 数据输入与输出.....	23
2.3 Python 程序中的运算符与表达式.....	25
2.3.1 算术运算符	26

2.3.2	比较运算符	27
2.3.3	赋值运算符	28
2.3.4	逻辑运算符	29
2.3.5	按位运算符	30
2.3.6	成员运算符	31
2.4	Python 中的数据类型	32
2.4.1	整数类型	33
2.4.2	浮点数类型	33
2.4.3	复数类型	35
2.4.4	布尔类型	35
2.5	Python 内置的运算函数	36
2.5.1	内置的数值运算函数	37
2.5.2	内置的字符串处理函数	39
第 3 章	控制程序执行流程	43
3.1	结构化程序设计	43
3.1.1	结构化流程图	44
3.1.2	程序运行的基本结构	45
3.2	顺序结构	46
3.2.1	计算圆的面积 S 和周长 L	47
3.2.2	计算正方形的面积 S	48
3.3	分支结构	49
3.3.1	单分支结构	49
3.3.2	双分支结构	52
3.3.3	多分支结构	54
3.3.4	if 语句嵌套结构	57
3.3.5	多重条件判断	59
3.4	循环结构	60
3.4.1	遍历循环 (有限循环): for 语句	60
3.4.2	无限循环 (条件循环): while 语句	62
3.4.3	循环辅助语句: break 和 continue	63

第 4 章 Python 中函数的使用	66
4.1 函数的定义和基本使用	66
4.1.1 函数的定义	66
4.1.2 函数的调用	67
4.1.3 lambda 表达式的使用	67
4.2 参数传递	68
4.2.1 理解形式参数和实际参数	69
4.2.2 位置实参	70
4.2.3 关键字实参	72
4.2.4 默认值	73
4.3 变量的作用域	73
4.3.1 局部变量	73
4.3.2 全局变量	74
4.4 函数的递归	75
4.4.1 递归的定义	75
4.4.2 递归的使用方法	76
4.5 函数模块	76
4.5.1 导入模块	77
4.5.2 导入函数	77
4.5.3 用 as 指定别名	78
第 5 章 组合数据类型	79
5.1 序列	79
5.1.1 索引	80
5.1.2 切片	81
5.1.3 序列相加	81
5.1.4 乘法	82
5.1.5 检查某个元素是否是序列的成员	83
5.1.6 计算序列的长度、最小值和最大值	83
5.2 列表	84
5.2.1 创建列表	84
5.2.2 访问列表中的元素	85

5.2.3	列表中的常见函数	85
5.3	元组	87
5.3.1	创建元组	87
5.3.2	访问元组中的元素	88
5.3.3	修改元组变量	89
5.4	集合	90
5.4.1	创建集合	90
5.4.2	集合处理函数	91
5.4.3	集合的操作	92
5.5	字典	93
5.5.1	字典的创建	93
5.5.2	访问字典的值	94
5.5.3	字典中的常用函数	95
第 6 章	文件与文件系统	96
6.1	文件的打开和关闭	96
6.1.1	文件的打开	96
6.1.2	文件的关闭	99
6.1.3	文件的读/写操作	100
6.2	文件和目录操作模块	105
6.3	常见的目录及文件操作	106
6.3.1	路径的获取	106
6.3.2	判断目录是否存在	107
6.3.3	创建、删除和修改目录	107
6.3.4	文件的重命名	110
第 7 章	正则表达式	113
7.1	概述	113
7.2	基本元字符	114
7.3	正则表达式在 Python 中的使用	115
7.4	re 模块中的常用函数及其功能	116
7.4.1	match()、search()、findall()函数	116

7.4.2	compile()函数	119
7.4.3	split()函数	119
7.4.4	sub()与subn()函数	120
7.5	分组匹配和匹配对象	122
7.5.1	分组匹配	122
7.5.2	匹配对象	122
第8章	程序进程和线程	124
8.1	进程	124
8.1.1	认识进程	125
8.1.2	通过Process类创建进程	125
8.1.3	通过继承Process类创建进程	126
8.2	线程	127
8.2.1	认识线程	128
8.2.2	thread模块	128
8.2.3	threading模块	131
8.2.4	线程同步	134
第9章	Python操作数据库	136
9.1	Python数据库接口和API	136
9.1.1	通用接口和API	136
9.1.2	连接对象和游标对象	137
9.2	Python操作关系型数据库	138
9.2.1	关系型数据库简介	138
9.2.2	用Python操作MySQL数据库	139
9.3	Python操作非关系型数据库	147
9.3.1	非关系型数据库简介	147
9.3.2	Python操作XML	148
9.4	Python操作嵌入式数据库	149
9.4.1	嵌入式数据库简介	149
9.4.2	Python操作SQLite数据库	150

第 10 章 Web 网站编程技术.....	153
10.1 Flask 框架.....	153
10.1.1 Flask 框架简介.....	153
10.1.2 Flask 框架的安装.....	154
10.1.3 Flask 框架的第一个程序.....	155
10.1.4 Flask 框架的应用.....	155
10.2 Django 框架.....	162
10.2.1 Django 框架简介.....	162
10.2.2 Django 框架的安装.....	162
10.2.3 使用 Django 框架创建 HelloWorld 项目.....	162
10.2.4 Django 框架的应用.....	163
第 11 章 Python 可视化编程.....	169
11.1 NumPy 库概述.....	169
11.2 使用 NumPy 库.....	170
11.2.1 数据的维度和 NumPy 库.....	170
11.2.2 NumPy 库的使用详解.....	171
11.3 图像的手绘效果.....	183
11.3.1 图像的数组表示.....	183
11.3.2 图像的手绘效果实现.....	186
11.4 使用 matplotlib 库.....	188
11.4.1 matplotlib.pyplot 库简介.....	188
11.4.2 matplotlib.pyplot 库深入.....	191
11.5 综合案例 1: 绘制极坐标图.....	197
11.6 综合案例 2: 绘制股票 K 线图.....	198
第 12 章 基于 PyQt 5 技术的 GUI 编程.....	201
12.1 GUI 概述.....	201
12.1.1 GUI 是什么.....	201
12.1.2 GUI 工具集.....	202
12.2 PyQt 简介及安装.....	203
12.2.1 PyQt 简介.....	203

12.2.2	PyQt 安装	203
12.3	Qt Designer 入门	205
12.3.1	Qt Designer 简介	205
12.3.2	Qt Designer 安装	205
12.3.3	窗口的基本介绍	206
12.3.4	.ui 文件的查看及与.py 文件的转换	208
12.4	PyQt 5 基本窗口空间的使用方法	209
12.4.1	Qt 中的 3 个基类	209
12.4.2	Qt 中的主要控件	211
12.5	GUI 的布局管理	213
12.5.1	绝对位置布局	213
12.5.2	布局类简介	214
12.5.3	布局类进阶	214
第 13 章	使用 Python 开发交互式游戏	217
13.1	Pygame 简介与安装	217
13.1.1	检测是否安装 Pygame 模块	218
13.1.2	安装 pip	218
13.1.3	安装 Pygame 模块	221
13.2	开发 Pygame 游戏	222
13.2.1	第一个 Pygame 游戏	222
13.2.2	熟悉 Pygame 游戏最小框架	223
13.3	制作接弹球计分游戏初始部分	225
13.3.1	游戏概述	225
13.3.2	创建 Pygame 窗口及游戏初始化	226
13.3.3	美化游戏界面	228
13.3.4	添加游戏弹球	229
13.3.5	添加接球球拍	230
13.4	制作接弹球计分游戏主体部分	232
13.4.1	让弹球动起来	232
13.4.2	建立弹球与球拍的关联性	234
13.4.3	增加游戏的难度	235

13.5	优化游戏弹球运动速度	238
第 14 章	智能机器人——利用 Python 智能写诗	239
14.1	认识智能化	239
14.1.1	智能化的概念	239
14.1.2	Python 在智能化时代的编程优势	240
14.2	系统设计	240
14.2.1	系统功能结构	240
14.2.2	系统效果预览	241
14.3	系统开发必备	241
14.3.1	系统的开发环境	241
14.3.2	项目文件结构	242
14.4	窗体 UI 的设计和实现	242
14.4.1	窗体 UI 概述	242
14.4.2	配置 Qt Designer	243
14.4.3	在 Qt Designer 中设计窗体	245
14.4.4	将.ui 文件转换为.py 文件	246
14.5	第三方库介绍	247
14.5.1	random 库	247
14.5.2	choice 库	247
14.5.3	os.path 库	247
14.5.4	第三方库在 PyCharm 中的安装	247
14.6	程序的实现过程	250
14.6.1	窗体的构建过程	251
14.6.2	运行时显示窗体	252
14.6.3	写诗代码的实现过程	252
14.7	程序的运行及运行结果	255
14.7.1	程序的运行	255
14.7.2	古诗词的文本库	255
14.7.3	程序的运行结果	256

第 15 章 人工智能——人脸识别系统	257
15.1 需求分析.....	257
15.2 系统设计.....	258
15.2.1 系统功能结构.....	258
15.2.2 系统效果预览.....	258
15.3 系统开发必备.....	259
15.3.1 开发工具介绍.....	259
15.3.2 项目文件结构.....	260
15.4 系统功能的实现.....	260
15.4.1 项目文件的创建.....	260
15.4.2 项目所需模块的安装.....	262
15.4.3 程序的设计.....	264
15.4.4 程序的输出.....	270
第 16 章 数据可视化——天气预测系统	271
16.1 需求分析.....	271
16.2 系统设计.....	271
16.2.1 系统功能结构.....	272
16.2.2 系统效果预览.....	272
16.3 系统开发必备.....	274
16.3.1 开发工具准备.....	274
16.3.2 模块介绍.....	274
16.3.3 项目文件结构.....	275
16.4 系统功能的实现.....	276
16.4.1 项目文件的创建.....	276
16.4.2 项目所需模块的安装.....	278
16.4.3 程序的设计.....	279
16.4.4 将数据存入数据库.....	284

1

第 1 章

认识 Python 语言

当下，无论是大数据、人工智能，还是机器学习，Python 都是首选语言。本章具体讲解了程序设计语言的基础，如程序设计语言的分类，编译和解释，Python 的由来、优缺点、应用领域及发展，Python 程序开发环境配置、运行，Python 解释器和集成开发环境，以及程序运行流程，为后续学习打下坚实的基础。

本章重点知识：

- Python 基础知识。
- Python 程序开发环境的建立。
- Python 解释器与安装 PyCharm。

1.1 走进Python

Python 是目前非常热门的编程语言，崇尚优雅、明确、简单。它继承了传统编译语言的强大功能和通用性，同时借鉴了简单脚本和解释语言的易用性。

Python 是由吉多·范罗苏姆（Guido van Rossum，人称“龟叔”，见图 1-1）于 1989 年年底发明的，第一个公开发行人版发行于 1991 年。像 Perl 语言一样，Python 源代码同样遵循 GPL（GNU General Public License，通用公共授权协议）标准。

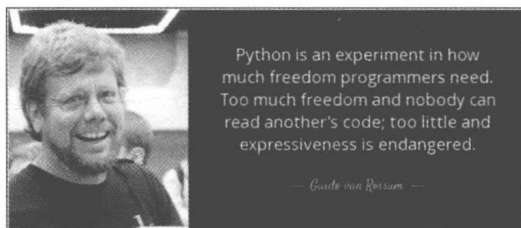


图 1-1 Python 创始人像

用 Python 编写的程序具有很强的可读性，相比其他语言，它的语法结构非常有特色。

- Python 是解释性语言：这意味着在开发过程中没有编译这个环节，类似于 PHP 和 Perl 语言。
- Python 是交互式语言：这意味着可以通过一个 Python 提示符来直接互动执行应用程序。
- Python 是面向对象语言：这意味着 Python 支持面向对象的风格或代码封装在对象中的编程技术。
- Python 是初学者的语言：Python 对初级程序员而言是一种伟大的语言，它支持广泛的应用程序开发，从简单的文字处理到 WWW 浏览器，再到游戏。

1.1.1 Python 的优缺点

通过上面的介绍，可以了解到 Python 是一门动态解释性语言。那么，这门语言具有哪些优缺点呢？

1. Python 的优点

Python 具有如下优点。

1) 易学

Python 的定位是“优雅”“明确”“简单”，所以 Python 程序看上去非常简单易懂。初学者学习 Python 很容易入门，而且将来深入下去，可以编写那些非常复杂的程序。

2) 开发效率高

Python 拥有强大的第三方库，基本上常用的计算机能实现的功能，在 Python 官方库里都有相应的模块提供支持。直接下载调用后，在基础库的基础上再进行开发，可以大大缩短开发周期，避免重复“造轮子”。

3) 高级语言

在用 Python 编写程序的时候，无须考虑诸如如何管理程序所使用的内存等底层细节。