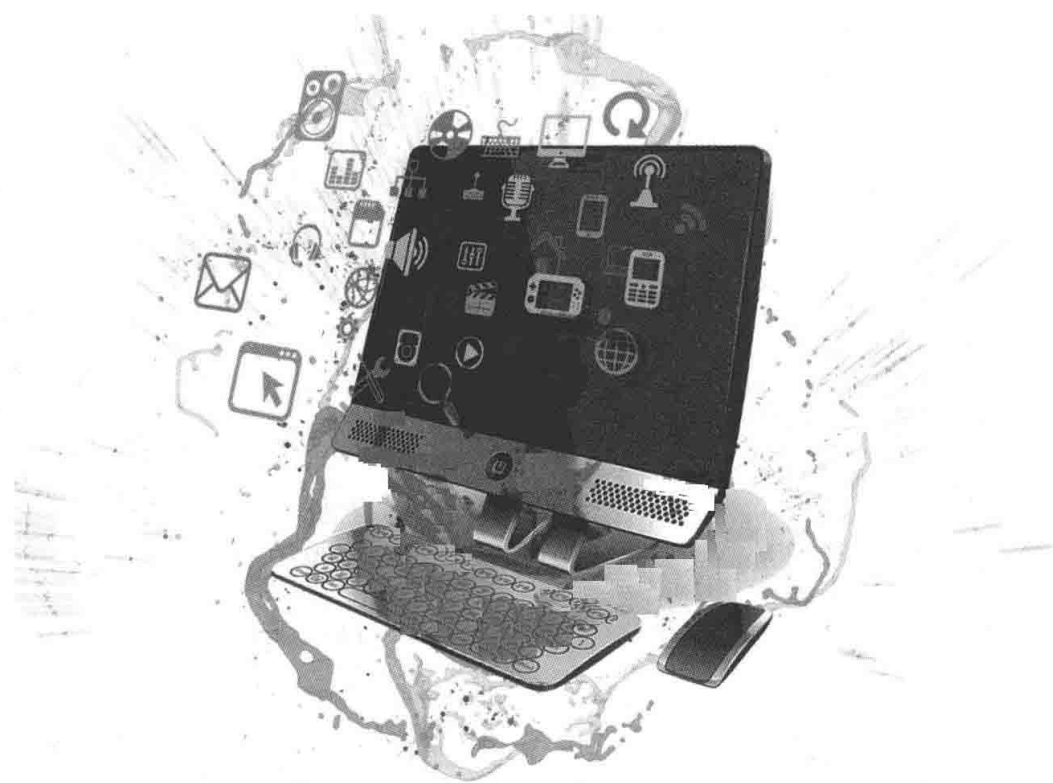


Python

开发技术大全

吴仲治◎编著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

Python开发技术大全/吴仲治编著. —北京: 机械工业出版社, 2020.8

ISBN 978-7-111-66152-8

I. P… II. 吴… III. 软件工具—程序设计 IV. TP311.561

中国版本图书馆CIP数据核字 (2020) 第132404号

Python 开发技术大全

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 李华君

责任校对: 姚志娟

印 刷: 中国电影出版社印刷厂

版 次: 2020 年 8 月第 1 版第 1 次印刷

开 本: 186mm×240mm 1/16

印 张: 35.25

书 号: ISBN 978-7-111-66152-8

定 价: 149.00 元

客服电话: (010) 88361066 88379833 68326294

投稿热线: (010) 88379604

华章网站: www.hzbook.com

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光/邹晓东

Python 作为一门编程语言，有 30 多年的历史了。其热度在最近几年迅速攀升，究其原因，除了因为该语言具有简洁、容易上手的特点外，还因为它在人工智能、数据处理和爬虫等多个领域提供了非常优秀的开发库。这些优秀的 Python 库极大地扩展了 Python 的应用领域，并且使得 Python 逐步成为这些领域的首要开发语言。本书兼顾 Python 的语法知识和 Python 在各个领域的应用实例，不但能够让读者快速入门，而且可以让读者比较全面地掌握 Python 编程中的一些高级主题。

笔者将多年的知识积累和务实的工作经验浓缩成这本书，奉献给千千万万学习 Python 编程的人。本书包含大量的代码实例，涉及 Python 语言的方方面面。本书讲解深入浅出、通俗易懂，将抽象的概念用典型实例浅显直白地进行了讲解。本书涉及的内容繁多，不但全面涵盖 Python 语言的基础语法，而且涵盖多线程、多进程、Redis、容器、爬虫、Flask 框架、图形界面应用和数据处理等诸多高级主题。不论哪方面的内容，本书都强调实战，而绝不囿于示意与演示。读者学到的将会是紧跟时代发展的技术，是完全可以在实际工作中直接使用的开发知识。相信借助本书，你的 Python 学习之旅一定会成为一种难忘的体验。

本书特色

1. 内容全面、实用，讲解详略得当，章节安排符合学习者的认知规律

本书全面涵盖 Python 编程的相关知识，所讲解的内容非常注重实用性和可操作性。书中对必须掌握的细节绝不吝惜笔墨，都是通过真实的代码演示各个知识点；而对仅需要大致了解的内容也绝不浪费纸张，这使得图书的整体结构紧凑，内容重点突出。这样的安排能够帮助读者快速上手，并且将精力重点放在培养解决实际问题的能力上。这种由浅入深、循序渐进的讲授方式完全遵循和尊重学习者尤其是初学者对软件开发的认知规律。

2. 涵盖大量生动有趣的实例，帮助读者深刻理解所学知识

本书中的每个章节都包含一些生动有趣的实例，通过人们在生活中经常碰到的一些例子来演示 Python 编程的知识点。这些例子都是我们在学习其他编程语言时经常会碰到的一些经典实例，例如如何让 3 种动物安全过河、汉诺塔问题和背包问题等。笔者在书中详细介绍了这些例子的编码实现过程，并对一些重点和难点问题做了分析。

在介绍 Python 知识时，笔者的讲授绝不是那种死板教条的模式，而是从一个程序员的视角对这些知识进行生动有趣、通俗易懂的讲解。这犹如一位编程伙伴陪伴你在 Python

主题公园中游玩，让你在不知不觉中就对 Python 编程有了全面的了解，帮助你在较短的时间内成为 Python 编程达人。全面阅读本书后，你将会发现整个学习过程并不痛苦，而且还非常有趣。

3. 涉及面很广，不仅适合Python初学者，也适合Python进阶者

本书从最基本的 Python 语言历史和 Python 基础语法开始讲解，逐步深入到诸如类、多线程、多进程、容器、爬虫和数据处理等高级话题，不仅适合 Python 初学者学习，让他们快速跨入 Python 编程的大门，而且适合 Python 进阶开发者阅读，让他们全面掌握 Python 编程涉及的高级话题，解决实际应用开发过程中碰到的种种难题。

本书内容

第1篇 Python编程基础（第1~8章）

第 1 章从 Python 的历史入手，介绍 Python 的版本、Python 语言的特点及 Python 开发需要的基础知识，掌握这些内容可以为后续章节的学习打好基础。

第 2 章介绍整型、浮点数、字符串、布尔型、列表、元组、集合和字典等 Python 定义的基本数据类型，以及这些数据类型的运算操作。我们将会在后续章节中使用这些基本数据类型构造自己的数据类型。

第 3 章介绍函数的定义和使用、函数参数的使用、常见的系统自带函数，以及 lambda 函数和递归等方面的内容。

第 4 章介绍模块和包的概念与定义，另外还介绍第三方 Python 包的各种安装方法。

第 5 章介绍文件操作的方方面面，包括打开文件、读写文件、文件模式和 with 语句的使用等。

第 6 章介绍正则表达式、数学库、时间库、日期库、二叉树、串行化、随机数等常用模块，这些模块涵盖了 Python 开发过程中的基础内容。

第 7 章介绍如何捕捉异常、得到异常信息、忽略某些异常、抛出异常、自定义异常对象，以及异常处理在 Python 2 和 Python 3 中的异同，最后介绍常见的系统定义异常。

第 8 章介绍类及其属性和方法、类的派生方法、多重派生的使用等内容。通过类可以定义自己的类型，从而可以不仅仅使用系统定义的类型。

第2篇 Python编程高级主题（第9~21章）

第 9 章介绍线程的概念和特点、使用线程类创建和停止线程、线程同步、线程安全等内容。通过使用线程，程序可以获得并发能力。

第 10 章介绍进程的概念和特点、使用进程类创建和停止进程、查看进程状态、在进程之间共享数据、同步进程及协调进程的运行等内容。通过进程可以使用其他可执行文件

提高代码的性能。

第 11 章介绍迭代器和生成器的概念，以及如何使用迭代器节省系统资源，如何操作生成器获得期望的数据等。

第 12 章介绍装饰器的起源、装饰函数与被装饰函数的参数、常用装饰器的用法等。

第 13 章介绍如何处理 csv 文件、yaml 文件和 json 文件等。这些都是常见的数据文件格式，可以使用它们保存数据，也可将其用于不同程序之间的数据交换。

第 14 章介绍软件测试的概念、PyUnit 和 Pytest 的使用、生成测试报告、产生覆盖率数据、运行自动化测试用例、共享测试脚手架等内容。程序代码必须经过测试才可以发布，借助测试框架可以大幅度提升代码的质量。

第 15 章介绍网络编程的概念、socket 接口、UDP 客户端程序的编写、UDP 服务器程序的编写、TCP 客户端程序的编写、TCP 服务器程序的编写、socket 的属性控制等。通过本章的学习，可以编写出基于 TCP/IP 协议的网络应用程序。

第 16 章介绍容器的概念、容器镜像的下载、容器实例的启动、容器实例的查看、容器镜像的创建、容器镜像的发布、通过 kubernetes 发布应用等。通过本章的学习，可以使用自己编写的 Python 代码完成基于容器的运维工作。

第 17 章介绍爬虫的概念、爬虫的工作原理、Scrapy 框架的安装和使用、通过过滤器来提取感兴趣的内容、配置 Scrapy 并爬取指定范围的页面、将爬取的数据保存成所希望的文件格式、使用模板、以交互式运行爬虫等。通过本章的学习，可以掌握从互联网上爬取各种有用数据的方法。

第 18 章介绍 Python 的 Web 开发框架 Flask 的工作原理、路由、通用参数和模板的使用等。通过本章的学习，能够构造自己的 Web 服务器，并提供 REST 服务或 Web 服务。

第 19 章介绍 Redis 键值数据库的概念、Redis 服务器的搭建、Redis 客户端库、元操作、字符串操作、哈希数据的操作、列表和集合的操作、订阅机制、时效设置等内容。通过本章的学习，可以搭建 Redis 服务器，操作 Redis 服务器上的数据，并通过 Redis 服务器进行数据交换。

第 20 章介绍 Tkinter、wxPython、PyQt 等图形界面库的使用，以及如何使用这些库提供的元素，另外还介绍图片的使用、用户操作事件的处理等内容。通过本章的学习，可以编写出自己的图形界面程序。

第 21 章介绍 matplotlib 的使用、各种统计数据图的生成、NumPy 数据处理、NumPy 金融应用、使用 NumPy 进行傅里叶变换、使用 NumPy 训练神经网络等。通过本章的学习，可以编写出各种处理工程数据的程序，还可以训练自己的神经网络。

配书资源

本书涉及的所有实例源代码文件及其他相关资源需要读者自行下载。请在华章公司的网站 www.hzbook.com 上搜索到本书，然后单击“资料下载”按钮，即可在本书页面上找

到配书资源下载链接。

读者对象

- Python 初学者与进阶者；
- Python 从业人员；
- Python 编程爱好者；
- 互联网产品开发人员；
- 数据运维人员；
- 数据清洗与分析人员；
- 数据库管理人员；
- 桌面应用开发人员；
- 高校计算机等专业的学生；
- 培训机构的学员。

本书作者

本书由吴仲治编写。笔者是一个拥有近 30 年软件开发经验的老兵，使用过 Fortran、Pascal、Perl、C/C++、C#和 Java 等高级语言，在电信、互联网、云计算等领域有丰富的开发经验。近十年笔者逐步转到 Python 开发，目前正在享受 Python 带来的快速开发体验，以及 Python 在各领域的各种库所带来的便捷。作为一个 Python 语言的深度用户，笔者愿意将自己多年的 Python 开发经验分享给广大 Python 编程爱好者，以帮助他们快速成长。

本书涉及的内容比较庞杂，而且 Python 技术也是日新月异，加之编者水平和成书时间所限，书中难免存有一些疏漏和不当之处，敬请指正。阅读本书时若有疑问，请发 E-mail 到 hzbook2017@163.com。

前言

第 1 篇 Python 编程基础

第 1 章 Python 语言基础	2
1.1 Python 语言介绍	2
1.2 Python 版本介绍	3
1.2.1 Python 语言的版本	3
1.2.2 Python 的实现版本	4
1.3 集成开发环境	9
1.3.1 PyCharm	9
1.3.2 Anaconda	10
1.4 代码的运行方式	17
1.5 网页运行方式	19
1.6 基本语法	23
1.7 Python 之禅	26
第 2 章 基本数据类型	28
2.1 整数类型	28
2.1.1 整数的表示法	28
2.1.2 数值运算	29
2.1.3 位运算	31
2.1.4 移位运算	33
2.1.5 类型转换	34
2.1.6 其他进制表示法	35
2.1.7 小整数池	39
2.1.8 统计成绩的例子	40
2.1.9 韩信点兵的故事	41
2.2 布尔类型	42
2.2.1 布尔运算	42
2.2.2 类型转换	44
2.2.3 例子：谁在说谎	44

2.3	字节流	46
2.4	浮点数	50
2.4.1	基本运算	51
2.4.2	其他运算	52
2.4.3	类型转换	53
2.4.4	例子：求某个数的平方根	53
2.5	列表	54
2.5.1	列表特性	55
2.5.2	常见操作	56
2.5.3	找出丢失的数	60
2.5.4	背包问题	61
2.6	字符串	63
2.6.1	str 类型的字符串	64
2.6.2	str 类型的相关接口函数	64
2.6.3	bytes 类型	68
2.6.4	str 类型和 bytes 类型的转换	69
2.7	元组	70
2.7.1	元组的表示法	70
2.7.2	元组接口函数	71
2.8	集合	72
2.8.1	创建集合	72
2.8.2	基本操作	73
2.8.3	例子：删除重复元素	79
2.9	字典	80
2.9.1	创建字典	81
2.9.2	基本操作	81
2.9.3	例子：合并结果	86
2.9.4	例子：如何安全过河	87
2.10	NoneType 类型	94
2.11	变量、对象和值的关系	94
第 3 章	函数	99
3.1	函数的定义和使用	99
3.2	函数的返回值	100
3.3	函数的参数	100
3.3.1	位置参数	101
3.3.2	将位置参数当作元组	101
3.3.3	调用时使用元组和列表	103

3.3.4	关键字参数	104
3.3.5	将实参当作字典	104
3.3.6	调用时使用字典	105
3.3.7	默认值	105
3.4	常见的系统自带函数	106
3.4.1	数据对象有关的函数	106
3.4.2	数学计算有关的函数	109
3.4.3	容器类型有关的函数	111
3.4.4	属性操作	116
3.5	lambda 函数	117
3.6	递归调用	118
3.6.1	递归调用介绍	118
3.6.2	例子	119
第 4 章	模块和包	126
4.1	模块	126
4.2	包	128
4.3	软件包的安装	129
4.3.1	复制源代码到系统目录	129
4.3.2	使用 PIP 进行安装	130
4.3.3	使用 easy_install 和 setup 进行安装	134
第 5 章	文件操作	135
5.1	文件的基本操作	135
5.2	打开文件	136
5.3	文件对象的操作	137
5.4	关闭文件	142
5.5	特殊文件	143
5.6	文件对象的属性	145
第 6 章	基础库	146
6.1	正则表达式库	146
6.1.1	元字符和语法	147
6.1.2	C 风格接口函数	154
6.1.3	面向风格接口函数	155
6.1.4	re.Match 对象	155
6.2	数学库	157
6.3	时间库	159
6.3.1	时间表示法	159

6.3.2	接口函数	160
6.4	日期库	162
6.5	二叉树库	162
6.6	串行化库	164
6.6.1	基本流程	164
6.6.2	pickle 用法	165
6.7	随机数库	166
6.7.1	生成随机整数和随机实数	167
6.7.2	其他形式的随机	169
6.7.3	伪随机问题	170
6.7.4	例子：计算面积	171
6.7.5	游戏：五对胡	172
第 7 章	异常处理	177
7.1	Python 中的异常处理	177
7.1.1	捕捉异常	177
7.1.2	没有异常才执行的分支	180
7.1.3	无论有无异常都要执行的分支	181
7.2	抛出异常	182
7.3	常见的系统定义异常	183
第 8 章	类和对象	187
8.1	类的定义和实例创建	187
8.2	类的属性	189
8.3	初始化函数和销毁函数	190
8.4	实例对象的属性	192
8.4.1	类属性和实例对象属性的关系	192
8.4.2	self 在类定义中表示当前对象	194
8.4.3	通过 delattr()、setattr() 等接口函数来操作属性	196
8.5	静态方法	198
8.6	类方法	199
8.7	访问控制	199
8.8	只读属性	200
8.8.1	使用@property 修饰符	200
8.8.2	重新定义__setattr__()函数	201
8.9	类属性的安全检查	202
8.10	派生类	203
8.11	多重继承	204

第 2 篇 Python 编程高级主题

第 9 章 多线程	208
9.1 线程介绍	208
9.2 Python 中的线程	209
9.2.1 主线程	209
9.2.2 线程状态	209
9.3 thread 包	210
9.3.1 线程的启动	211
9.3.2 线程的退出	212
9.4 threading 包	213
9.4.1 创建线程实例对象	213
9.4.2 派生自己的线程类	216
9.4.3 配置线程	217
9.4.4 启动线程	218
9.4.5 停止线程	219
9.4.6 等待线程结束	220
9.4.7 其他接口函数	221
9.5 线程同步	222
9.5.1 线程锁 thread.Lock	223
9.5.2 线程锁 threading.Lock	224
9.5.3 可重入锁 threading.RLock	225
9.5.4 条件变量 threading.Condition	227
9.5.5 信号量 threading.Semaphore	228
9.5.6 事件 threading.Event	231
9.6 线程安全	233
9.7 死锁	236
第 10 章 多进程	239
10.1 进程的概念	239
10.2 subprocess 模块	240
10.2.1 面向过程的接口函数	241
10.2.2 Popen 类	244
10.3 multiprocessing 模块	247
10.3.1 创建进程	247
10.3.2 进程的属性	249
10.3.3 进程的接口函数	253

10.3.4	进程池	258
10.3.5	进程通信	260
10.4	os 模块	265
10.4.1	启动进程——system()	265
10.4.2	启动进程并得到输出——popen()	265
10.4.3	得到当前进程 ID——getpid()	266
10.4.4	得到父进程 ID——getppid()	266
第 11 章	迭代器和生成器	268
11.1	迭代器	268
11.2	生成器	269
11.2.1	定义生成器	270
11.2.2	接口函数	270
第 12 章	装饰器	275
12.1	起源	275
12.2	被装饰函数带有参数	276
12.3	装饰函数带有参数	279
12.4	装饰函数带有返回值	280
12.5	使用多个装饰器	281
12.6	常用装饰器	282
12.6.1	类装饰器	282
12.6.2	退出时执行的函数	284
12.6.3	单例模式	284
12.6.4	执行时间限制	285
12.6.5	执行时间标定	286
12.6.6	自动重新运行	287
12.6.7	状态机	288
第 13 章	数据文件操作	293
13.1	csv 文件的处理	293
13.2	yaml 文件的处理	296
13.3	json 文件的处理	298
第 14 章	软件测试	301
14.1	无框架的测试	301
14.2	PyUnit 测试框架	305
14.2.1	运行方式	307
14.2.2	生成测试报告	309
14.2.3	测试步骤	312

14.2.4	断言	317
14.2.5	测试用例的接口函数	328
14.2.6	覆盖率	330
14.3	Pytest 测试框架	333
14.3.1	测试用例	333
14.3.2	选择执行的测试用例	335
14.3.3	生成测试报告	339
14.3.4	测试步骤	341
14.3.5	断言	345
14.3.6	覆盖率	346
14.3.7	插件	349
第 15 章	网络编程	357
15.1	TCP/IP 协议介绍	357
15.2	socket 介绍	361
15.3	TCP echo 服务器	363
15.4	简单的 TCP 客户端	364
15.5	UDP 版的 echo 服务器	365
第 16 章	容器操作	367
16.1	Docker 的安装	367
16.2	Docker 的基本概念	369
16.3	docker-py 库	370
16.3.1	建立连接	370
16.3.2	对 Docker 镜像的操作	371
16.3.3	运行 Docker 镜像	377
16.4	kubernetes 库	379
第 17 章	Scrapy 爬虫框架	388
17.1	Scrapy 的安装	388
17.2	工作原理	390
17.3	一个简单的例子	391
17.4	Scrapy 的架构图	393
17.5	爬虫	395
17.5.1	点赞统计例子	395
17.5.2	爬虫类	396
17.6	一个完整的项目	398
17.7	结果数据	400
17.8	选择器	402

17.8.1	介绍	403
17.8.2	text()	404
17.8.3	extract()/extract_first()	405
17.8.4	@属性	406
17.8.5	元素[@属性]	406
17.9	设置	407
17.9.1	命令行方式	407
17.9.2	爬虫范围内的设置	408
17.9.3	项目范围内的设置	408
17.9.4	命令默认设定	409
17.9.5	全局默认设定	409
17.9.6	读取配置	411
17.9.7	常用配置	412
17.10	命令行工具	414
17.10.1	创建项目	414
17.10.2	创建爬虫	415
17.10.3	列出爬虫模板	416
17.10.4	查看某个爬虫模板	417
17.10.5	基于模板创建爬虫	418
17.10.6	启动爬虫	418
17.10.7	爬虫列表	419
17.10.8	爬取指定页面	420
17.10.9	在浏览器中显示页面	421
17.10.10	启动交互式 shell	422
17.10.11	runspider	423
17.10.12	标定 Scrapy 的性能	424
17.10.13	其他命令	425
17.11	爬取北京公交线路数据	426
第 18 章	Flask Web 开发框架	429
18.1	Flask 的安装	429
18.2	最简单的 Flask 应用	430
18.3	路由	430
18.3.1	动态路由	431
18.3.2	限定类型	432
18.3.3	地址尾部的 “/”	433
18.4	运行应用的参数	435
18.5	命令行方式运行	436

18.6	HTTP 的方法	437
18.7	静态文件	437
18.8	模板	438
18.8.1	变量	439
18.8.2	表达式	440
18.8.3	子模板	442
第 19 章	键值数据库 Redis	444
19.1	介绍	444
19.2	服务器的安装	445
19.2.1	服务器容器下载	446
19.2.2	启动 Redis 容器	446
19.3	Python 客户端的安装	448
19.4	元操作	448
19.5	字符串数据的操作	449
19.6	哈希数据的操作	456
19.7	列表数据的操作	458
19.8	集合数据的操作	460
19.9	订阅发布	462
19.9.1	基本用法	462
19.9.2	订阅多个频道	465
19.9.3	频道使用通配符	466
19.10	设置时效	467
19.11	管道模式	467
第 20 章	图形界面应用	469
20.1	Tk 和 Tkinter	469
20.1.1	基本程序结构	470
20.1.2	常用组件	471
20.1.3	布局	487
20.1.4	游戏 2048	489
20.2	wxPython	494
20.2.1	程序基本结构	494
20.2.2	基本用户界面元素	495
20.3	PyQt	508
20.3.1	基本程序结构	508
20.3.2	常用组件	509
20.4	小结	517

第 21 章 数据处理	518
21.1 matplotlib	518
21.2 NumPy	526
21.2.1 ndarray 的构造	526
21.2.2 ndarray 的一些属性	531
21.2.3 ndarray 的一些操作	532
21.2.4 NumPy 在金融领域的应用	534
21.2.5 NumPy 中的傅里叶变换	537
21.2.6 神经网络应用	541