

Python

快速入门精讲

邹琪鲜 编著



扫一扫
免费观看视频
获取全部源代码



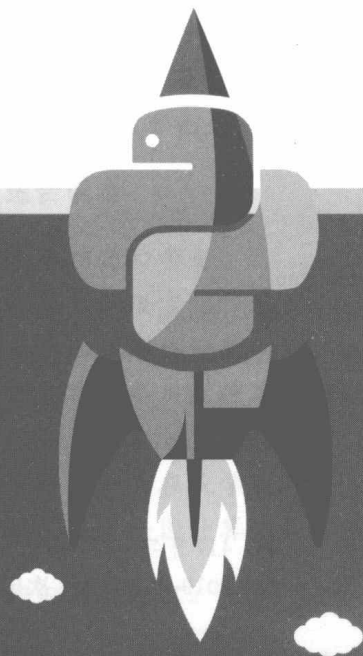
ZERO TO HERO

清华大学出版社

Python

快速入门精讲

邹琪鲜 编著



清华大学出版社
北京

内 容 简 介

本书从零基础入门着手,通过合理的编排,首先引导读者循序渐进地学习 Python 基本语法和语义,再掌握诸如文件和数据库的处理、面向对象编程、开发图形用户界面、网络和多线程编程等实用技术,最后拓展了 Python 的一些热门应用,如大数据和机器学习。

本书通过丰富的案例和真切的经验分享、详细的文字讲解和生动的在线视频演示,给读者带来别样的学习体验。

本书非常适合没有 Python 编程经验的程序员,也适合工作中需要用到程序解决问题的非专业人员,由于独特的编排和讲解,同样适合初学 Python 的学生,甚至可以当作全国计算机二级 Python 的教材使用。总之,对于第一次接触编程的人来说,这是一本非常适合的书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Python 快速入门精讲/邹琪鲜编著. —北京:清华大学出版社,2019
ISBN 978-7-302-51478-7

I. ①P… II. ①邹… 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 254885 号

责任编辑:王剑乔

封面设计:刘 键

责任校对:李 梅

责任印制:杨 艳

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:17.25

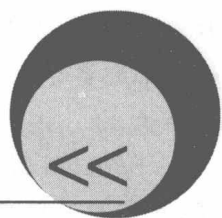
字 数:417千字

版 次:2019年3月第1版

印 次:2019年3月第1次印刷

定 价:49.00元

产品编号:077434-01



在这本书的创作过程中,有过很多的构思,是精炼直接还是面面俱到,是道理连篇还是实用为主……经过深思熟虑,最终的呈现是:涉及技术的地方,用简练的语言去介绍,希望读者能够用最短的时间了解一个新功能或者一个陌生的领域;涉及思想认知方面,用生活化的语言进行详细说明。

读者还会发现本书在介绍一些基础知识的时候,会有一些“废话”。对于有编程经验的人来讲,确实是废话;而对于一个新手来讲,我希望这些“废话”是一个过来人的经验分享,就好像有个老师在你面前娓娓道来,而不是几句话就涵盖很多的概念和知识点,让新手可能想破头也想不通这里的来龙去脉。我希望通过这样的方式能够帮助新手树立正确的编程观念,继而产生潜移默化的影响。

至于这本书,我希望它是一个指引,而不是什么宝典大全。当读过之后,我希望读者能脱离这本书,形成自己的思维方式和解决问题的方法,而不是总来翻这本书寻找答案。书中不单讲解语言本身,更是注重培养读者的编程意识,通过经验的分享和抛出问题的方式培养读者独立探索和解决新问题的能力,这也是一个合格的程序员所需要具备的素质。

适合读者

零基础的初学者。

掌握其他语言的程序员。

需要编程解决问题的运维工程师。

工作中需要用到程序解决问题的非计算机领域工作者。

编程爱好者。

本科及大专在校学生。

想要报考全国计算机等级考试的读者。

如何学习本书

本书共 3 篇 18 章,分别为基础入门篇、进阶应用篇和拓展案例篇。

基础入门篇为第 0 章至第 7 章,内容从克服编程恐惧开始,逐步渗透编程思想,建立编程的信心。知识从 Python 安装开始,以合理的递进式知识结

构编排,介绍了 Python 的数据类型、语句语法以及函数和模块化编程,每个章节的重要知识点都配备了生动的教学视频,可以采取文字和视频同步学习的方式,能更快、更好地吸收新知识。

进阶应用篇为第 8 章至第 15 章,主要介绍文件和数据持久化、面向对象编程、异常处理、开发图形用户界面、正则表达式以及爬虫入门、多线程编程和网络应用编程等相关实用技能,在其中还适当穿插了一些自学任务。通过这部分学习,可以让读者对 Python 的应用有更多了解,增强自学能力,达到学一通三的效果,从而可以在自己的实际工作中快速学习 Python 相关的、新的模块和功能并加以应用。各章节均配备了视频教程。

拓展案例篇是第 16 章至第 18 章,每章通过一个案例讲解某一个领域的应用。因为 Python 的特殊性,Python 在很多领域都有所应用,作为新手可能没办法在短时间内精通 Python 在所有领域的应用。这里选取了三个案例,分别是大数据、语音识别和机器学习入门,每个案例都可以帮助新手对相关领域快速了解并入门,如果对哪个方面感兴趣,或者在工作中有需要,可以再深入进行学习研究,这几个案例都是非常好的入门指南。

另外,在本书中还有一些特别的设计,比如,在基础入门和进阶应用部分有一个叫《英雄无敌》的游戏项目,这其实并不是教你开发游戏的教程,只是通过一种载体让那些看起来无关的语法或者功能有一个有机的结合,对零散知识点与程序设计有一个系统的认知。随着学习的深入,你可以让游戏的功能越来越完善,甚至超出书中的设计,这个游戏可以从最简单的文字形式发展成游戏界面;从单纯的文字到具备复杂故事情节;可以从单机到网络。总之,只要你学了新的知识就可以想想这个功能可以在游戏中用来干什么。在书里很多地方我都做了引导,如果你有想法,就去实现吧!

在本书的“动动手”部分也留了一些开放式的问题,需要自己去考虑和设计,有些做了视频演示,如果想不通,也可以到 QQ 交流群中跟大家进行探讨。

至于实验环境,书中主要的案例都是基于 Python 3+Windows 10 的环境,个别拓展内容会涉及 Linux 和 Python 2,不过请放心,一定会在你的理解范围内。

相关资源

书中提供了丰富的代码演示,所有的代码都可以在 Github 得到,Github 网址是 <https://github.com/milozou/CrazyPython/>。代码按章分 18 个目录保存。

如果想跟更多人在线交流经验,可以加入疯狂的 Python QQ 交流群-1: 814674076。

如果有内容勘误方面的意见,欢迎直接发邮件至 zouqixian@gmail.com。

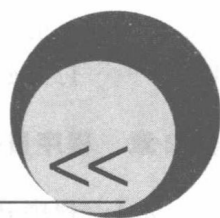
最后,希望大家在提到本书时,不会说这是一本 Python 编程的宝典,我希望大家会说这是一本武功心法。看完之后,醍醐灌顶,打通了任督二脉,内功深厚,哈哈。



本书配套源代码

邹琪鲜

2019 年 1 月



基础入门篇

第 0 章 从零开始	3
0.1 克服编程恐惧	3
0.2 如何写出好程序	4
0.3 为什么选择 Python	5
0.4 Python 的发展和应用	6
0.5 一些建议	8
0.6 多平台搭建 Python 开发环境	10
0.6.1 什么是开发环境	10
0.6.2 获得 Python 安装包	11
0.6.3 安装 Python	12
0.7 命令行模式及 Python 的第一次运行	13
第 1 章 开始编程	19
1.1 第一个程序的诞生	19
1.1.1 编程动机	19
1.1.2 神奇的导入: import	19
1.1.3 画一个五角星	21
1.1.4 Python 对话	22
1.1.5 编写程序	23
1.2 熟悉开发环境,提高编程效率	25
1.3 Python 开发工具	25
1.4 第三方模块和工具管理	26
1.5 像程序员一样写代码	28
1.5.1 注释	28
1.5.2 模块导入	33

1.5.3	表达式和语句	33
1.5.4	合理利用空白	33
第2章	程序员基础	38
2.1	程序开发全局观	38
2.2	数据的标签:变量	40
2.2.1	声明变量	40
2.2.2	变量名和值的关系	42
2.3	编写可以跟用户互动的程序:输入、处理和输出	44
2.4	快速理解对象和类型:数字和字符串	46
2.5	运算符和表达式	47
2.5.1	算术运算符	47
2.5.2	赋值运算符	48
2.5.3	比较运算符	49
2.5.4	逻辑运算符和布尔值	49
2.5.5	成员运算符	49
2.5.6	其他运算符	50
2.5.7	运算符优先级	50
2.6	如何快速获取帮助	51
2.7	彩蛋:打印正弦波	52
第3章	搞定字符串	55
3.1	字符串的基本定义	55
3.1.1	转义字符	56
3.1.2	Docstring	57
3.1.3	原始字符串	57
3.1.4	Unicode 字符串	58
3.2	序列	58
3.2.1	索引	58
3.2.2	切片	59
3.3	与字符串相关的运算符	60
3.3.1	拼接和重复	60
3.3.2	比较运算符	61
3.3.3	成员判断	62
3.4	灵活多变的字符串操作	63
3.4.1	函数	63
3.4.2	对象和方法	63
3.4.3	分割和拼接	66
3.4.4	字符串模块	67

3.5	字符串格式化	68
3.6	遍历字符串	70
第 4 章	流程控制	72
4.1	让程序变智能的分支结构:if 语句	73
4.1.1	if 语法结构	73
4.1.2	布尔值与 if	75
4.1.3	逻辑运算符与 if	76
4.2	条件循环:while 语句	77
4.2.1	while 语句	77
4.2.2	while...else 语句	79
4.2.3	死循环和 break	80
4.2.4	确定次数的循环	80
4.3	迭代循环:for 语句	81
4.3.1	容器和迭代器	82
4.3.2	实例:斐波那契数列	84
4.3.3	循环嵌套	85
4.3.4	循环控制 continue	85
第 5 章	列表和元组	87
5.1	《英雄无敌》迭代开发:构建英雄世界	87
5.2	程序中的数据仓库:列表	88
5.2.1	创建列表	88
5.2.2	列表拆分	89
5.3	列表的序列化操作	89
5.3.1	索引和切片	89
5.3.2	运算符及函数	90
5.3.3	遍历	91
5.4	列表的操作	92
5.4.1	可变的列表	92
5.4.2	列表的方法	93
5.4.3	字符串和列表	94
5.5	Python 的魔术	94
5.5.1	列表推导式	95
5.5.2	生成器表达式	95
5.5.3	一点建议	96
5.6	深拷贝、浅拷贝	96
5.6.1	赋值	96
5.6.2	浅拷贝	98

5.6.3	深拷贝	99
5.7	不可变的列表——元组	100
5.7.1	创建元组	100
5.7.2	元组赋值	101
5.7.3	列表和元组	101
5.7.4	什么时候使用元组	102
5.8	《英雄无敌》需求落地	102
第 6 章	分治策略——函数与模块	105
6.1	函数基础	105
6.1.1	自定义函数	105
6.1.2	形参和实参	106
6.1.3	返回值	107
6.2	变量作用域	108
6.2.1	局部变量	108
6.2.2	全局变量	109
6.2.3	命名空间	109
6.3	参数的类型	110
6.3.1	默认参数	110
6.3.2	关键参数	110
6.3.3	冗余参数处理	111
6.3.4	序列和字典做实参	112
6.4	内建函数	112
6.5	匿名函数:lambda 表达式	114
6.6	生成器 yield 语句	115
6.7	模块和包	116
6.7.1	模块	117
6.7.2	导入模块	117
6.7.3	搜索路径	118
6.7.4	包	118
6.7.5	__name__ 属性	119
第 7 章	字典和集合	121
7.1	字典	121
7.1.1	创建字典	121
7.1.2	字典的键和值	122
7.1.3	字典的相关操作	122
7.1.4	字典的方法	123
7.2	字典实例:统计高频词	124

7.3	字典的妙用	125
7.4	集合	126
7.4.1	Python 集合	126
7.4.2	集合的方法和应用	127



进阶应用篇

第 8 章	文件和数据持久化	133
8.1	文件读取	133
8.2	文件写入	136
8.3	文件内的指针	137
8.4	文件关闭	138
8.5	文件名和路径	138
8.6	os 模块	140
8.7	捕获异常	141
8.8	数据序列化	142
8.8.1	pickle 模块	142
8.8.2	json 模块	143
8.9	CSV 文件	144
8.9.1	CSV 模块	144
8.9.2	CSV 读写	145
第 9 章	面向对象	148
9.1	从《英雄无敌》开始认识对象	148
9.2	从面向过程到面向对象	150
9.3	类和对象	151
9.4	属性和方法	152
9.4.1	类的属性	153
9.4.2	类的方法	154
9.4.3	内置属性和方法	155
9.5	类的继承	156
9.5.1	使用继承	156
9.5.2	重载	158
9.6	多态	160
9.7	内置装饰器	160
9.8	《英雄无敌》面向对象设计	162
第 10 章	异常处理	165
10.1	异常	165

10.2	Python 的异常类	166
10.3	捕获和处理异常	167
10.3.1	try...except...语句	167
10.3.2	try...except...else 语句	168
10.3.3	finally 子句以及嵌套	169
10.3.4	谁都跑不了	169
10.4	抛出异常	170
10.4.1	raise 语句	170
10.4.2	自定义异常类	171
10.4.3	assert 语句	171
第 11 章	开发图形用户界面	173
11.1	GUI	173
11.2	tkinter	174
11.2.1	创建空白窗口	174
11.2.2	添加组件	175
11.2.3	事件绑定	176
11.2.4	其他组件	176
11.3	wxPython	177
11.3.1	子类化开发: 空白窗口	178
11.3.2	添加组件及窗口布局	179
11.3.3	事件绑定	182
11.3.4	布局管理器	184
11.4	GUI 可视化构建工具: 用 wxFormBuilder 开发 GUI 程序	187
11.5	生成可执行的二进制文件	193
第 12 章	Python 玩转数据库	195
12.1	数据库初始	195
12.2	SQLite 数据库	196
12.3	Python 连接 MySQL	198
第 13 章	分身有术: 多线程编程	201
13.1	进程与线程	201
13.2	多线程	202
13.2.1	创建线程	202
13.2.2	线程对象的方法	204
13.2.3	线程锁	205
13.2.4	多线程的本质	207
13.3	实例: 批量主机扫描	207

第 14 章 网络应用编程	210
14.1 网络应用开发	210
14.2 socket 编程	211
14.2.1 socket 连接过程	211
14.2.2 创建 socket 对象	212
14.2.3 基于 TCP 的客户端和服务端	213
14.2.4 基于 UDP 实现多线程收发消息	215
14.3 实例:局域网聊天室	216
14.3.1 需求分析	216
14.3.2 概要设计	217
14.3.3 详细设计	217
14.3.4 编码阶段	219
第 15 章 正则表达式	225
15.1 正则表达式的常用字符	225
15.1.1 普通字符	225
15.1.2 元字符	226
15.2 Python 中的 re 模块	229
15.2.1 正则表达式主要功能	229
15.2.2 re 模块使用的两种形式	229
15.2.3 re 常用函数及方法	229
15.3 实例:一只小爬虫	232
拓展案例篇	
第 16 章 小白也玩大数据	241
16.1 好玩的大数据	241
16.2 大数据技术	242
16.3 MapReduce 模型	243
16.4 案例:实现 MapReduce 模型	243
16.4.1 案例设计	243
16.4.2 分割文件	243
16.4.3 编写 map 函数	244
16.5 彩蛋:词云	249
第 17 章 语音识别技术	252
17.1 选择语音识别包	252

17.2	speech 模块	253
17.2.1	语音识别开发环境搭建	253
17.2.2	环境配置和调试	254
17.2.3	文字和声音相互转化	255
17.2.4	speech 模块的其他方法	256
第 18 章	六行代码入门机器学习	257
18.1	人工智能发展简史	257
18.2	机器学习初体验:搭建机器学习环境	258
18.3	机器学习的过程	260
18.3.1	收集训练数据	260
18.3.2	训练分类器并做出预测	261
参考文献	263

基础入门篇

- 第 0 章 从零开始
- 第 1 章 开始编程
- 第 2 章 程序员基础
- 第 3 章 搞定字符串
- 第 4 章 流程控制
- 第 5 章 列表和元组
- 第 6 章 分治策略——函数与模块
- 第 7 章 字典和集合

万事开头难,学习编程似乎也不容易。对于新手来讲,在眼花缭乱的资讯、教程、分享以及五花八门的资源中,首先需要的就是选对学习路线。有太多从入门到放弃的例子,就是因为没有找到学习的主线,把时间精力都浪费在无用的知识和信息上,最后看起来好像学了很多东西,但是真正有用的也就是十分之一。

学习 Python 编程不同于其他计算机技术,并不是罗列出一堆知识点,然后逐一记住它们,也不是同样功能的几个方法都要掌握。很多文章或书籍都会堆砌知识点,相同功能也要列出几个,这就好像孔乙己在说“回”字有几种写法一样。新手初看,感觉内容很丰富,学着学着就没动力了,真的是体验了“学海无涯”。而高手往往会觉得拼凑、罗列知识点的方式过于累赘。其实,学习编程把最主要的核心学会,再掌握解决问题的正确思路,就已经算是入门了,Python 更是如此。

这是个崇尚“最好的办法只有一个”的语言,无须学那么多,遇到需要学习的新功能,运用已经掌握的方法,很快就可以上手并运用。这也是一个合格程序员应该掌握的能力。



疯狂的 Python

0.1 克服编程恐惧

新手学习编程之前通常会有两种情绪:好奇和恐惧。通常小孩子或者纯小白对编程的好奇心会比较大,好奇是因为觉得编程很神奇,可以让计算机帮助我们完成更多的工作;而顾虑比较多的人就会产生恐惧或者畏难心理,恐惧是因为对未知的迷茫和无序。比如很多人在开始学习之前会先说“我的英语不好”“我的数学不好”等类似的问题。

Python 可以说是最适合编程入门的语言了,少儿接触编程,Python 也是最好的选择。既然如此,那就说明这门语言本身其实是非常简单的。

其实,对于新手来说,首先要搞明白的事情就是到底什么是编程,以及明确编程的困难之处到底在哪里。

学编程就好像学习写文章,不同的是,文章是写给人看的,程序是写给计算机看的。我们都有过学习写文章的经历,那是一个过程:从陌生到熟悉的过程,从笨拙到巧妙的过程。没有人是刚认识字就可以写出漂亮文章的。

想想我们学习写作文的过程,最开始的时候并不是一上来老师就跟你说:“来,我们现在学习写作文。”也不是先学汉字或者拼音,我们首先学会的是把想法用语言表达出来,也

就是说,我们最先会的是在脑子里形成想法,那可能是下意识的,也可能是经过深思熟虑产生的想法。有了想法之后,我们可以通过口述表达出来,然后才逐渐开始学习拼音、汉字、词、句子、短语、段落、文章,造句写文章的规矩和语法以及不同类型文章的结构等。

学好中文就可以写出中文给懂中文的人看,而编程就是把想法变成程序给计算机看。先会说话后写作,学习编程也是一样,首先是要有想法,而不是先想要写什么代码。有了想法之后,就是怎么把想法通过程序实现出来。编程就是把想法变成代码的过程。互联网行业的很多创业者都有一个共同的困境,那就是“我有一个好想法,就差一个程序员了”。有了 Python,想法变程序就容易得多了。

至于学习编程的困难之处,让我们想想在学习除了母语之外另一门语言的过程吧。或许你会说“我的英语很好”“我的法语很厉害”,那么,如果用这门外语写诗呢?想想我们的古诗,要合辙押韵,还有更难的八股文,你能想象用法语写古诗、英语写八股文吗?感受到难度在哪儿了吗?古诗跟普通白话文章的区别就在于,古诗有字数限制,有韵脚的要求。而学习一门外语并且用外语写诗面临两个问题:

- (1) 学习一门陌生的语言。
- (2) 按照这门语言的语法和韵脚写诗。

编程要面临的困难也是这两个问题:

- (1) 不知道或不熟悉编程语言的语法和语义。
- (2) 不知道如何让计算机解决问题(就好像不知道怎么用法语写诗一样)。

所以,编程的困难就在于,要学习一门跟母语不一样的语言,而且为了计算机能明白,还要学习一些特定的语法和规则;同时还要用这个语言去解决问题。

明确了编程的困难,接下来就是怎样克服困难。学习编程最好的办法就是动手不断地编程,就跟学习所有的东西一样。想要做一个好的守门员,就要熟悉球性,不断练习各种基本动作(语法)、熟悉规则(编码规范)、研究团队策略(算法)等。其中最好的做法就是多练习、多实战。

最后,你需要明确编程的两个层次:

- (1) 找到解决问题的方法。
- (2) 编写出好程序。

0.2 如何写出好程序

出色大牛的程序员和菜鸟程序员的差别就在于能不能写出好程序。在我们知道了编程语言(比如 Python)的语法和语义后,怎样才能写出好程序呢?

关于什么是好程序,其定义是仁人见仁智者见智,跟行业内的朋友们聊天,特别是带团队的 CTO 们,经常会听到各种吐槽。比如:

- (1) 明明写的屎粑粑一样的代码,还自我感觉良好。
- (2) 写代码都不过脑子,上来就敲,还吹嘘自己一天写了多少行,写得再多,也只是造了更多的垃圾。
- (3) 瞎设计,有简单的方法不用,还自以为高明。你提醒他,他还觉得自己有个性。
- (4) 走路都没走稳,就想着灵活,灵活设计是给小白的吗?不是,那是给高手出招的,