

水木书荟

理解为主，应用为王

案例引导，快乐学习

Python
轻松入门

第一版畅销13万册

实用·好玩·参与

零基础 入门学习 Python



立体化教程
1800
分钟视频

(第2版) 微课视频版 © 小甲鱼 著

清华大学出版社



水木书荟

零基础 入门学习 Python



(第2版) 微课视频版 © 小甲鱼 著



清华大学出版社
北京

内 容 简 介

本书提倡理解为主，应用为王。因此，只要有可能，小甲鱼（注：作者）都会通过生动的实例来让大家理解概念。

虽然这是一本入门书籍，但本书的“野心”并不止于“初级水平”的教学。本书前半部分首先讲解基础的 Python 3 语法知识，包括列表、元组、字符串、字典以及各种语句；之后循序渐进地介绍一些相对高级的主题，包括抽象、异常、魔法方法以及属性迭代器。后半部分则围绕着 Python 3 在爬虫、界面开发和游戏开发上的应用，通过实例引导读者进行深入学习和探究，既富有乐趣，又锻炼了读者的动手能力。

本书适合学习 Python 3 的入门读者，也适合对编程一无所知，但渴望用编程改变世界的朋友们。

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

零基础入门学习 Python：微课视频版 / 小甲鱼著. —2 版. —北京：清华大学出版社，2019
(2019.8重印)

(水木书荟)

ISBN 978-7-302-51408-4

I. ①零… II. ①小… 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 237486 号

责任编辑：刘 星
封面设计：刘 健
责任校对：胡伟民
责任印制：李红英

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

课 件 下 载：<http://www.tup.com.cn>, 010-62795954

印 装 者：清华大学印刷厂

经 销：全国新华书店

开 本：185mm×260mm

印 张：27.5

字 数：670 千字

版 次：2016 年 11 月第 1 版

2019 年 6 月第 2 版

印 次：2019 年 8 月第 3 次印刷

印 数：145001~155000

定 价：89.00 元

产品编号：081190-01

时光荏苒，一晃间，距离《零基础入门学习 Python》出版（2016 年 11 月）已经过去两年多了，在这段时间里，Python 逐步走入了大家的视野，这门语言因其简洁的语法风格，在云计算、金融分析、人工智能、科学运算和自动化运维等领域都有很好的应用，所以被越来越多的人所认识和接受，其使用率得到了大幅度的提升。

《零基础入门学习 Python》一经出版便受到了广大读者的欢迎，累计销售 13 万册，在出版后两年多的时间里，收到了很多读者朋友们的反馈，大部分的读者朋友给予了很高的评价，小甲鱼在此由衷地感谢大家。同时，也注意到朋友们提出的一些疑问、意见和建议。因此，在第 2 版中，小甲鱼对所使用的 Python 版本进行了更新（Python 3.7）；对书中存在的不足进行了弥补；引入了更多有趣的案例；添加了更多实用的模块讲解等。

本书更新和改进内容

(1) 所有案例均使用 Python 3.7 版本代替了原来的 Python 3.3，改写了大部分知识点的例子，使读者学习起来更富有趣味性。

(2) 考虑到现实中的开发场景，增加了一些案例：

- 在爬虫案例部分引入了流行的 Request 模块；
- 增加了“爬取豆瓣 Top250 电影排行榜”和“爬取网易云音乐的热门评论”案例；
- Scrapy 爬虫框架部分，采用了 Anaconda 来安装 Scrapy，使用 Scrapy 1.5.0 版本进行演示。

(3) 考虑到“正则表达式”和“Scrapy 爬虫框架”在实际开发中的应用非常广泛，将其从第 1 版中的第 14 章（论一只爬虫的自我修养）中独立出来，添加了更多的示例，使得内容更为翔实、丰富。

(4) 修改了第 1 版中的一些差错，在此要再次感谢各位读者提出的疑问，使小甲鱼能够发现书中的不足之处。

本书配套资源和网站支持

- PPT 课件请在清华大学出版社网站本书页面下载。
- 程序源代码和小甲鱼精心录制的 94 集（1800 分钟）视频教程，请扫描书中对应二维码获取。

注意：书中给出了下载程序源代码的二维码和视频观看二维码，请先扫描封四刮刮卡中的二维码进行注册（每个刮刮卡只能注册一个用户），之后再扫描相关二



维码即可获得配套资源。

- 同时，对于书中没有展开详述的内容提供了【扩展阅读】，读者可访问书中的相关网址或扫描对应位置的二维码进行阅读。部分原创的内容并不是免费提供的，读者可自行选择进行购买阅读。
- 本书还提供了额外的配套课后作业，如有需要，请在鱼 C 论坛 (<https://fishc.com.cn>) 或联系鱼 C 工作室的小客服 (<https://fishc.taobao.com>) 购买学习。
- 如果在学习中遇到困难，可以到鱼 C 论坛或关注鱼 C 工作室微信公众号获取相关知识，与各位网友们相互交流和讨论。论坛中的提问互助具有知识累积的特点，因为初学者很多问题是一样的，所以不妨在提问之前先在论坛搜索一下相关的关键词，一般都可以找到答案。

由于小甲鱼的水平有限，书中难免有一些错误和不准确的地方，恳请各位读者不吝指正，有兴趣的读者可发送邮件至 workemail6@163.com，期待收到大家的意见和建议。



鱼 C 工作室微信公众号



本书源代码和安装包下载
(含本书勘误)

我们一直在努力耕耘这么一片简单的土壤，虽然没有达到尽善尽美，但在大家的努力下，已初见雏形，并且在论坛上已经聚拢了很多超厉害的“大牛”！

Fake it till they make it —— 假装直到真的成功。

最后还是那句话，小甲鱼渴望和大家一起成长，十年前我们仰望星空，十年后我们将俯视大地。未来的天空，必将为我们留下一片灿烂的曙光！

小甲鱼

2019年3月

Life is short. You need Python.

——Bruce Eckel

上边这句话是 Python 社区的名言，翻译过来就是“人生苦短，我用 Python”。

我和 Python 结缘于一次服务器的调试，从此便一发不可收拾。我从来没有遇到一门编程语言可以如此干净、简洁。使用 Python，可以说是很难写出“丑陋”的代码。我从来没想过一门编程语言可以如此简单，它太适合零基础的朋友踏入编程的大门了，如果我有一个八岁的孩子，我一定会毫不犹豫地使用 Python 引导他学习编程，因为面对它，永远不缺乏乐趣。

Python 虽然简单，其设计却十分严谨。尽管 Python 可能没有 C 或 C++ 这类编译型语言运行速度那么快，但是 C 和 C++ 需要你无时无刻地关注数据类型、内存溢出、边界检查等问题。而 Python，它就像一个贴心的仆人，私底下为你都一一处理好，从来不用你操心这些，这让你可以将全部心思放在程序的设计逻辑之上。

有人说，完成相同的一个任务，使用汇编语言需要 1000 行代码，使用 C 语言需要 500 行，使用 Java 只需要 100 行，而使用 Python，可能只要 20 行就可以了。这就是 Python，使用它来编程，你可以节约大量编写代码的时间。

既然 Python 如此简单，会不会学了之后没什么实际作用呢？事实上并不担心这个问题，因为 Python 可以说是一门“万金油”语言，在 Web 应用开发、系统网络运维、科学与数字计算、3D 游戏开发、图形界面开发、网络编程中都有它的身影。目前越来越多的 IT 企业，在招聘栏中都有“精通 Python 语言优先考虑”的字样。另外，就连 Google 都在大规模使用 Python。

好了，我知道过多的溢美之词反而会使大家反感，所以我必须就此打住，剩下的就留给大家自己体验吧。

接下来简单地介绍一下这本书。2016 年，出版社的编辑老师无意间看到了我的一个同名的教学视频，建议我以类似的风格写一本书。当时我是受宠若惊的，也很兴奋。刚开始写作就遇到了不小的困难——如何将视频中口语化的描述转变为文字。当然，我希望尽可能地保留原有的幽默和风趣——毕竟学习是要快乐的。这确实需要花不少时间去修改，但我觉得这是值得的。

本书不假设你拥有任何一方面的编程基础，所以本书不但适合有一定编程基础，想学习 Python 3 的读者，也适合此前对编程一无所知，但渴望用编程改变世界的读者！本书提倡理解为主，应用为王。因此，只要有可能，都会通过生动的实例来让




大家理解概念。

编程知识深似海，没办法仅通过一本书所有的知识都灌输给你，但我能够做到的是培养你对编程的兴趣，提高你编写代码的水平，以及锻炼你的自学能力。


最后，本书贯彻的核心理念是：实用、好玩、参与。

小甲鱼


2016年7月

第 1 章 就这么愉快地开始吧..... 1 **视频讲解：13 分钟（1 个）**

- 1.1 获得 Python 1
- 1.2 从 IDLE 启动 Python 2
- 1.3 失败的尝试 3
- 1.4 尝试点儿新的东西 3
- 1.5 为什么会这样 4

第 2 章 用 Python 设计第一个游戏..... 5 **视频讲解：24 分钟（1 个）**

- 2.1 第一个小游戏 5
- 2.2 缩进 6
- 2.3 BIF 7

第 3 章 成为高手前必须知道的一些基础知识..... 9 **视频讲解：82 分钟（4 个）**

- 3.1 变量 9
- 3.2 字符串 10
- 3.3 原始字符串 11
- 3.4 长字符串 12
- 3.5 改进我们的小游戏 13
- 3.6 条件分支 14
- 3.7 初识循环 15
- 3.8 引入外援 16
- 3.9 闲聊数据类型 17
 - 3.9.1 整型 18
 - 3.9.2 浮点型 18
 - 3.9.3 布尔类型 19
 - 3.9.4 类型转换 19
 - 3.9.5 获得关于类型的信息 20
- 3.10 常用操作符 21
 - 3.10.1 算术操作符 21
 - 3.10.2 优先级问题 22



3.10.3 比较操作符	23
3.10.4 逻辑操作符	23

第4章 了不起的分支和循环 25

视频讲解：42 分钟（3 个）

4.1 分支和循环	25
4.2 快速上手	27
4.3 避免“悬挂 else”问题	28
4.4 条件表达式（三元操作符）	29
4.5 断言	31
4.6 while 循环语句	31
4.7 for 循环语句	32
4.8 break 语句	34
4.9 continue 语句	35
4.10 else 语句	35

第5章 列表、元组和字符串 37

视频讲解：137 分钟（7 个）

5.1 列表：一个“打了激素”的数组	37
5.1.1 创建列表	37
5.1.2 向列表添加元素	38
5.1.3 从列表中获取元素	39
5.1.4 从列表删除元素	40
5.1.5 列表切片	42
5.1.6 进阶玩法	43
5.1.7 一些常用操作符	44
5.1.8 列表的小伙伴们	46
5.2 元组：戴上了“枷锁”的列表	47
5.2.1 创建和访问一个元组	47
5.2.2 更新和删除元组	49
5.3 字符串	50
5.3.1 各种内置方法	51
5.3.2 格式化	54
5.4 序列	57

第6章 函数 62

视频讲解：140 分钟（8 个）

6.1 Python 的乐高积木	62
6.1.1 创建和调用函数	62



6.1.2	函数的参数	63
6.1.3	函数的返回值	64
6.2	灵活即强大	65
6.2.1	形参和实参	65
6.2.2	函数文档	66
6.2.3	关键字参数	67
6.2.4	默认参数	67
6.2.5	收集参数	68
6.3	我的地盘听我的	70
6.3.1	局部变量	70
6.3.2	全局变量	71
6.3.3	global 关键字	73
6.3.4	内嵌函数	73
6.3.5	LEGB 原则	74
6.3.6	闭包	75
6.3.7	装饰器	78
6.4	函数式编程	80
6.4.1	lambda	80
6.4.2	filter()	81
6.4.3	map()	82
6.5	递归	83
6.5.1	递归是什么	83
6.5.2	写一个求阶乘的函数	85
6.5.3	一帮小兔子——斐波那契数列	87
6.5.4	汉诺塔	89
第 7 章 字典和集合		92

 视频讲解：46 分钟（3 个）

7.1	字典：当索引不好用时	92
7.1.1	创建和访问字典	93
7.1.2	各种内置方法	95
7.2	集合：在我的世界里，你就是唯一	99
7.2.1	创建集合	100
7.2.2	访问集合	101
7.2.3	不可变集合	101

第 8 章 永久存储 103

 视频讲解：68 分钟（4 个）

8.1	文件：因为懂你，所以永恒	103
-----	--------------	-----



8.1.1	打开文件	103
8.1.2	文件对象的方法	104
8.1.3	文件的关闭	105
8.1.4	文件的读取和定位	105
8.1.5	文件的写入	106
8.1.6	一个任务	107
8.2	文件系统：介绍一个高大上的东西	109
8.3	pickle：腌制一缸美味的泡菜	115

第9章 异常处理 117

视频讲解：38分钟（3个）

9.1	你不可能总是对的	117
9.2	try-except 语句	120
9.3	try-finally 语句	122
9.4	raise 语句	122
9.5	丰富的 else 语句	123
9.6	简洁的 with 语句	124

第10章 图形用户界面入门 126

视频讲解：14分钟（1个）

10.1	安装 EasyGui	126
10.1.1	导入 EasyGui	127
10.1.2	快速入门	127
10.1.3	各种功能演示	129
10.2	默认参数和关键字参数	130
10.3	使用按钮组件	131
10.4	如何在 buttonbox 里边显示图片	132
10.5	为用户提供一系列选项	133
10.6	让用户输入消息	134
10.7	让用户输入密码	137
10.8	显示文本	138
10.9	目录与文件	139
10.10	捕获异常	140
10.11	记住用户的设置	141

第11章 类和对象 143

视频讲解：78分钟（5个）

11.1	给大家介绍对象	143
11.2	对象=属性+方法	143



11.3	面向对象编程	145
11.3.1	self 是什么	145
11.3.2	听说过 Python 的魔法方法吗	146
11.3.3	公有和私有	146
11.4	继承	148
11.4.1	调用未绑定的父类方法	150
11.4.2	使用 super 函数	151
11.5	多重继承	151
11.6	组合	152
11.7	类、类对象和实例对象	153
11.8	到底什么是绑定	154
11.9	一些相关的 BIF	156

第 12 章 魔法方法 160

 视频讲解：140 分钟（9 个）

12.1	构造和析构	160
12.1.1	__init__(self[, ...])	160
12.1.2	__new__(cls[, ...])	161
12.1.3	__del__(self)	162
12.2	算术运算	163
12.2.1	常见的算术运算	163
12.2.2	反运算	166
12.2.3	一元操作符	167
12.3	简单定制	167
12.4	属性访问	173
12.5	描述符（property 的原理）	177
12.6	定制序列	180
12.7	迭代器	182
12.8	生成器	185
12.9	生成器表达式	187

第 13 章 模块 190

 视频讲解：36 分钟（3 个）

13.1	模块就是程序	190
13.2	命名空间	191
13.3	导入模块	191
13.4	__name__ = '__main__'	192
13.5	搜索路径	194
13.6	包	195



13.7	像个极客一样去思考.....	196
第 14 章 论一只爬虫的自我修养..... 202		
视频讲解：76 分钟（4 个）		
14.1	入门.....	202
14.2	什么是编码.....	205
14.3	下载一只猫.....	205
14.4	更好的选择.....	207
14.4.1	没有对比就没有伤害.....	207
14.4.2	安装 Requests.....	208
14.4.3	安装 BeautifulSoup4.....	208
14.5	爬取豆瓣 Top250 电影排行榜.....	209
14.6	爬取网易云音乐的热门评论.....	213
第 15 章 正则表达式..... 221		
视频讲解：55 分钟（3 个）		
15.1	re 模块.....	221
15.2	通配符.....	221
15.3	反斜杠.....	222
15.4	字符类.....	223
15.5	重复匹配.....	223
15.6	特殊符号及用法.....	225
15.7	元字符.....	227
15.8	贪婪和非贪婪.....	229
15.9	反斜杠+普通字母=特殊含义.....	230
15.10	编译正则表达式.....	231
15.11	实用的方法.....	233
第 16 章 Scrapy 爬虫框架..... 238		
视频讲解：56 分钟（2 个）		
16.1	环境搭建.....	238
16.1.1	安装 Miniconda.....	238
16.1.2	安装 Scrapy.....	241
16.2	Scrapy 框架架构.....	241
16.3	创建一个 Scrapy 项目.....	243
16.4	编写爬虫.....	244
16.5	爬.....	245
16.6	取.....	246
16.6.1	在 Shell 中尝试 Selector 选择器.....	246



16.6.2	使用 XPath 选择器	247
16.6.3	使用 CSS 选择器	248
16.6.4	提取数据	249
16.6.5	在爬虫中提取数据	250
16.7	存储内容	251
16.8	跟进链接	252

第 17 章 GUI 的最终选择: Tkinter 254

视频讲解: 295 分钟 (14 个)

17.1	Tkinter 之初体验	254
17.2	进阶版本	255
17.3	Label 组件	258
17.4	Button 组件	260
17.5	Checkbutton 组件	261
17.6	Radiobutton 组件	262
17.7	LabelFrame 组件	264
17.8	Entry 组件	265
17.9	Listbox 组件	272
17.10	Scrollbar 组件	275
17.11	Scale 组件	276
17.12	Text 组件	278
17.12.1	Indexes 用法	280
17.12.2	Mark 用法	283
17.12.3	Tag 用法	285
17.13	Canvas 组件	292
17.14	Menu 组件	298
17.15	Menubutton 组件	302
17.16	OptionMenu 组件	303
17.17	Message 组件	305
17.18	Spinbox 组件	306
17.19	PanedWindow 组件	306
17.20	Toplevel 组件	309
17.21	事件绑定	311
17.22	事件序列	312
17.22.1	type	313
17.22.2	modifier	314
17.23	Event 对象	314
17.24	布局管理器	316
17.24.1	pack	316



17.24.2	grid	318
17.24.3	place	319
17.25	标准对话框	321
17.25.1	messagebox	321
17.25.2	fieldialog	323
17.25.3	colorchooser	324

第 18 章 Pygame: 游戏开发 326

视频讲解: 460 分钟 (19 个)

18.1	安装 Pygame	326
18.2	初步尝试	327
18.3	解惑	330
18.3.1	什么是 Surface 对象	330
18.3.2	将一个图像绘制到另一个图像上是怎么回事	330
18.3.3	移动图像是怎么回事	330
18.3.4	如何控制游戏的速度	331
18.3.5	Pygame 的效率高不高	332
18.3.6	应该从哪里获得帮助	332
18.4	事件	332
18.5	提高游戏的颜值	336
18.5.1	显示模式	336
18.5.2	全屏才是王道	337
18.5.3	使窗口尺寸可变	338
18.5.4	图像的变换	338
18.5.5	裁剪图像	341
18.5.6	转换图片	346
18.5.7	透明度分析	346
18.6	绘制基本图形	351
18.6.1	绘制矩形	351
18.6.2	绘制多边形	352
18.6.3	绘制圆形	353
18.6.4	绘制椭圆形	354
18.6.5	绘制弧线	355
18.6.6	绘制线段	356
18.7	动画精灵	356
18.7.1	创建精灵	358
18.7.2	移动精灵	360
18.8	碰撞检测	361
18.8.1	尝试自己写碰撞检测函数	362



18.8.2	sprite 模块提供的碰撞检测函数	365
18.8.3	实现完美碰撞检测	367
18.9	播放声音和音效	367
18.10	响应鼠标	371
18.10.1	设置鼠标的位置	371
18.10.2	自定义鼠标光标	372
18.10.3	让小球响应光标的移动频率	374
18.11	响应键盘	376
18.12	结束游戏	377
18.12.1	发生碰撞后获得随机速度	377
18.12.2	减少“抖动”现象的发生	378
18.12.3	游戏胜利	381
18.12.4	更好地结束游戏	382
18.13	经典飞机大战	389
18.13.1	游戏设定	389
18.13.2	主模块	390
18.13.3	我方飞机	392
18.13.4	响应键盘	393
18.13.5	飞行效果	394
18.13.6	敌方飞机	395
18.13.7	提升敌机速度	396
18.13.8	碰撞检测	398
18.13.9	完美碰撞检测	401
18.13.10	一个 BUG	402
18.13.11	发射子弹	403
18.13.12	设置敌机“血槽”	405
18.13.13	中弹效果	406
18.13.14	绘制得分	407
18.13.15	暂停游戏	408
18.13.16	控制难度	409
18.13.17	全屏炸弹	411
18.13.18	发放补给包	412
18.13.19	超级子弹	415
18.13.20	三次机会	417
18.13.21	结束画面	420

第1章

就这么愉快地开始吧



视频讲解

1.1 获得 Python

我观察到这么一个现象：很多初学的朋友都会在学习论坛上问什么语言才是最好的，他们的目的很明确，就是要找一门“最好”的编程语言，然后持之以恒地学习下去。没错，这种“执子之手，与子偕老”的专一精神是我们现实社会所推崇的。但在编程的世界里，我们并不提倡这样。我们更推崇“存在即合理”，当前热门的编程语言都有其存在的道理，它们都有各自擅长的领域和适用性。因此没办法通过某个单一的指标去衡量哪一门语言才是最好的。

Python 的语法非常精简，对于一位完美主义者来说，Python 将是他爱不释手的伙伴。Python 社区的目标就是构造完美的 Python 语言！本书将使用 Python 3 来进行讲解，而 Python 3 不完全兼容 Python 2 的语法，这样做无疑会让大多数程序员心生不满，因为他们用 Python 2 写的大量代码经过层层调试已经趋近完美，并已部署到成熟的生产环境中。对 Python 2 的不兼容，意味着他们需要将这些应用进行转换和重新调试，甚至重构……但是，Python 社区仍然坚持要舍弃 Python 2，推出全新的 Python 3。是的，只有勇敢地割掉与时代发展不相符的瑕疵部分，才能缔造出真正的完美体验！

“工欲善其事，必先利其器”。我们要成为“大牛”，要用 Python 去拯救世界，要做的第一件事就是下载一个 Python 的安装程序，并成功地将它安装到计算机上。

安装 Python 非常容易，可以在它的官网找到最新的版本并下载，地址是 <http://www.python.org>。

如图 1-1 所示，进入 Python 官网后找到 Download 字样，单击“Latest: Python 3.7.0”超链接，即可找到 Python 3.7.0 的下载地址。



注意：

本书使用的版本为 Python 3.7.0，通常情况下，只需要下载最新版本的 Python 3 即可，不影响学习。