

图灵电子书



编程小白的第1本
Python入门书

侯爵

写在前面：你需要这本书的原因

有没有哪一个瞬间，让你想要放弃学习编程？

在我决心开始学编程的时候，我为自己制定了一个每天编程1小时的计划，那时候工作很忙，我只能等到晚上9点，同事都下班之后，独自留在办公室编程。在翻遍了我能找到的几十本国内外的 Python 编程教程之后，我还是似懂非懂。那些教程里面到处都是抽象的概念、复杂的逻辑，对于专业开发者这些再平常不过，而对于我这样一个学设计出身的编程小白，没有被视觉化的东西是无法被理解的。

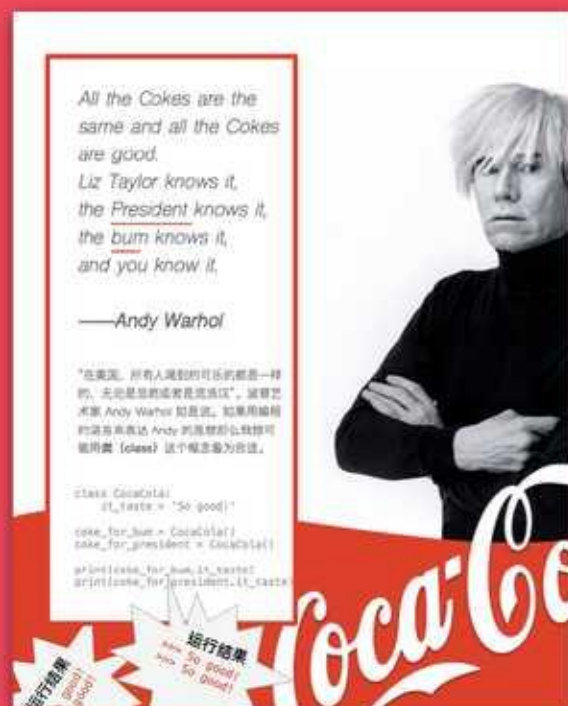
而且，这些书大多着重于一步步构建一个完整体系，但事实上，现实生活中没有哪个技能是这么习得的。难道要练习1年切菜才能给自己做一顿饭么？难道要到体校学习3年才能开始晨跑么？难道要苦练5年基本功才能开始拿起吉他弹第1首曲子么？

做任何事情一定有在短期内简单可行的方法。学习不应该是苦差事，而应该是快乐的，重要的是找到适合自己的学习方法。

既然笨办法不能让我学会 Python，那么我决定用一种聪明方法来学，为自己创造学习的捷径。这种高效学习法的核心在于：

- 1、**精简**：学习最核心的关键知识；
- 2、**理解**：运用类比、视觉化的方法来理解这些核心知识；
- 3、**实践**：构建自己的知识体系之后，再通过实践去逐渐完善知识体系。

实际上，如果你听说过《如何高效学习》中的整体学习法，你会发现它和我的高效学习法很相似，作者斯科特·杨用一年的时间学完了麻省理工四年的课程。既然这种高效学习法可以用来学习经济学、数学、物理，那么当然也可以用来学编程。



运用了高效学习法之后，我的编程学习速度突飞猛进，不到一个月时间就完成了爬虫加上 Django 的网站。为了追求最大程度的精简，我借用了许多强大的库和框架，这我不需要重新发明轮子，而是专注于最核心的功能。在一次项目合作中，我惊讶的发现，我用70行代码实现的功能，一名工程师竟然用了800行代码来实现。在运动场上，第一名和最后一名的差距也许只有几十秒。然而在编程这个竞技场上，不同方法的效率竟然会有10倍的差距，这也正是追求高效学习的价值所在。

为了能让更多的编程小白轻松地入门编程，我把高效学习法结合 Python 中的核心知识，写成了这本书。随意翻上几页，你就会发现这本书和其他编程书的不同，其中有大量的视觉化场景帮你理解概念，穿插了若干有趣的小项目，最重要的是，这本书是为零基础小白而设计的。

考虑到很多书都标榜是针对零基础读者的，所以有必要说明一下这一本有哪些独到之处。

我不写字典式的知识体系，有些编程书像字典一样，各种细枝末节的知识都涵盖在内。但我认为，正如不应该让小孩拿着字典学汉语，也不应该让初学者拿着这样的厚书学编程。在汉语中，掌握常见的1500-2000个字就能看懂80%的文字。在编程中，同样有着最核心的关键知识。先用这些关键的知识构建你的知识体系会让学习效率加速，这是比一开始就钻到各种细枝末节里面更好的学习方式。这是精简的意义。

我不去对比各种语言的特点，许多程序员背景的作者喜欢去对比 Python 和其他语言有什么异同，或者试图让你通过理解 C 语言从而理解 Python，但我不会这么做。我知道对于大多数读者，Python 很可能是将要学习的第一门编程语言，所以我不会用一个陌生概念讲解另一个陌生概念，反过来，**我会运用类比和视觉化的方法讲解 Python 中的抽象概念，把复杂的东西简单的讲清楚。**这是理解的捷径。

我不追求让你达到精通的程度，事实上我也很怀疑有哪本书能真正做到21天从入门到精通。精通一门语言，需要在实际项目开发中踩过许多坑，需要熟悉计算机运作的底层原理。我是一名实用主义的开发者，我相信你也一样，学习编程是为了真正做出点东西来，也许你想爬取大量的数据和信息，方便用来分析与决策。也许你想快速搭建一个网站，展示自己的产品。也许你对量化交易感兴趣，想试着把自己的投资策略程序化。对于实用主义的开发者来说，更应该追求的是“达成”而不是“精通”。**先掌握项目所需的最少必要知识，然后把热情和精力投入到搭建真实项目中，而不是死磕半年的基础知识，直到把所有兴趣都耗竭了也没做出来什么像样的东西。**在实践过程中，你自然会逐渐完善知识体系。在这本书里面，会穿插一些真实项目的片段，让你知道学了这个基础知识能用在哪儿，并且完成一些小型项目。这是让你最有成就感的实践。

说了这么多，就是为了让你能放下疑虑。这不是一本让你中途放弃的编程书，这是一本黏着你看完了的编程书。大多数读者都能在一周内读完，其中有35岁才开始学编程的中年男子、有工作非常忙碌的女性创业者、还有对编程感兴趣的高中生。所以，相信你也可以跟着这本书一起从零到一。

放轻松，如果你准备好了，那就翻开下一页吧。



作者介绍

麻瓜编程创始人。网易云课堂上最畅销的课程《Python 实战》系列课程讲师，目前已有超过4万名学员。

设计专业背景，拥有设计与编程跨界思维，善于找到学习技能的最佳路径，擅长把复杂的东西简单的讲清楚。

初学编程时，发现市面上很难找到适合小白的学习资料，于是开始用生动易懂、视觉化的方式来写这本教程。

目录

第一章 为什么选择 Python?	P7
第二章 现在就开始	P14
第三章 变量与字符串	P24
第四章 函数的魔法	P38
第五章 循环与判断	P54
第六章 数据结构	P79
第七章 类与可口可乐	P94
第八章 开始使用第三方库	P121
必读 给编程小白的学习资源	P131

第一章

为什么选择 Python ?

Why Python?

那些最好的程序员不是为了得到更高的薪水或者得到公众的仰慕而编程，他们只是觉得这是一件有趣的事情。

——Linux 之父 Linus Torvalds

作为一个实用主义的学习者，最关心的问题一定是「我为什么要选择学 Python，学会之后我可以用来做什么？」

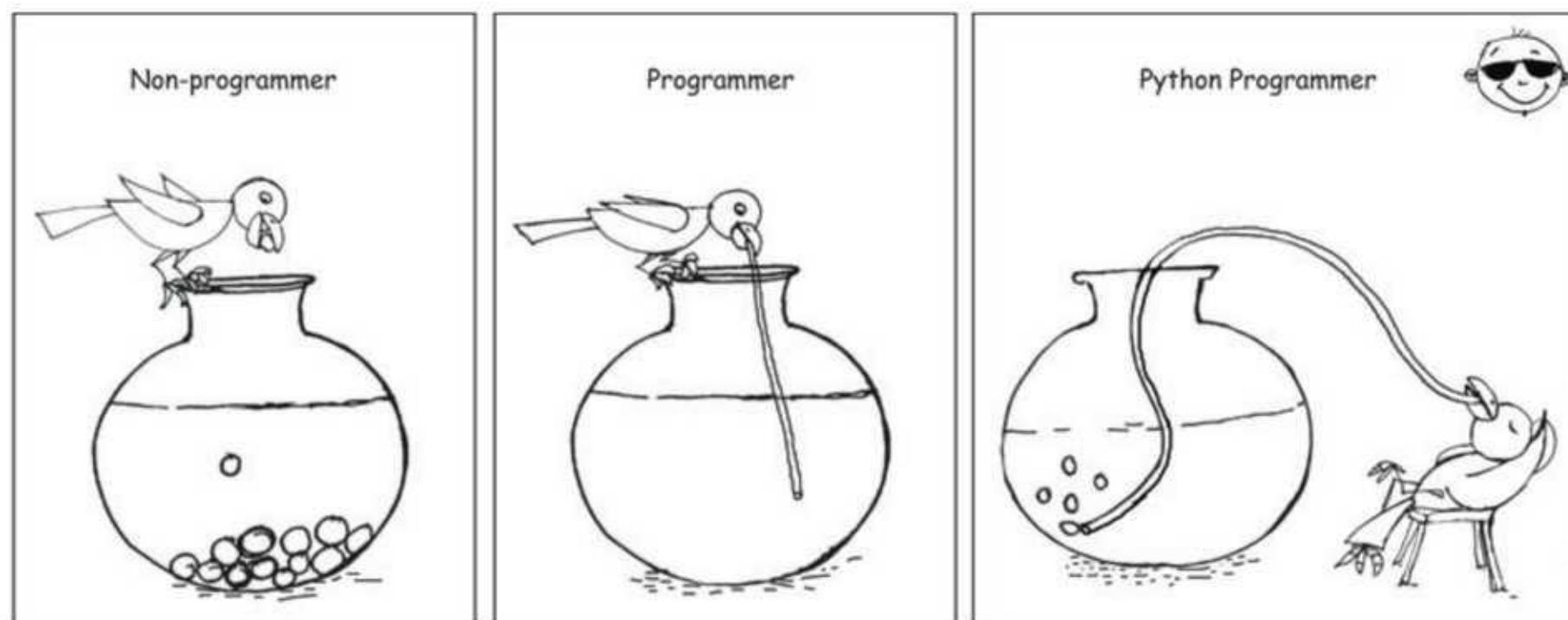
首先，对于初学者来说，比起其他编程语言，Python 更容易上手。

Python 的设计哲学是优雅、明确、简单。在官方的 The Zen of Python（《Python 之禅》）中，有这样一句话，

There should be one-- and preferably only one --obvious way to do it.

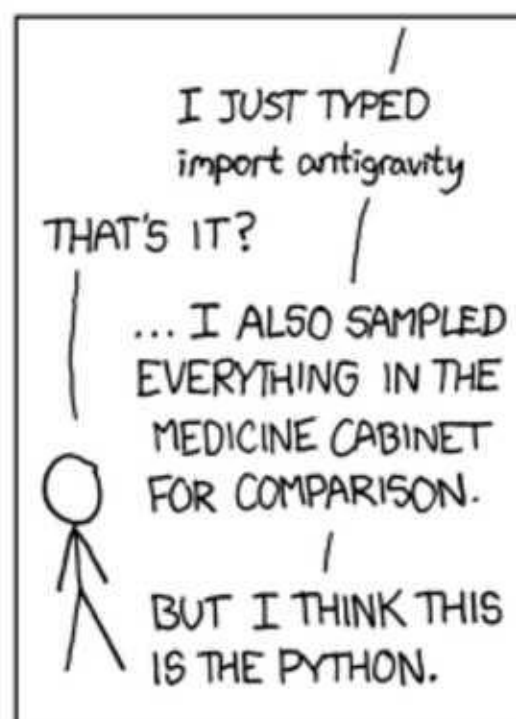
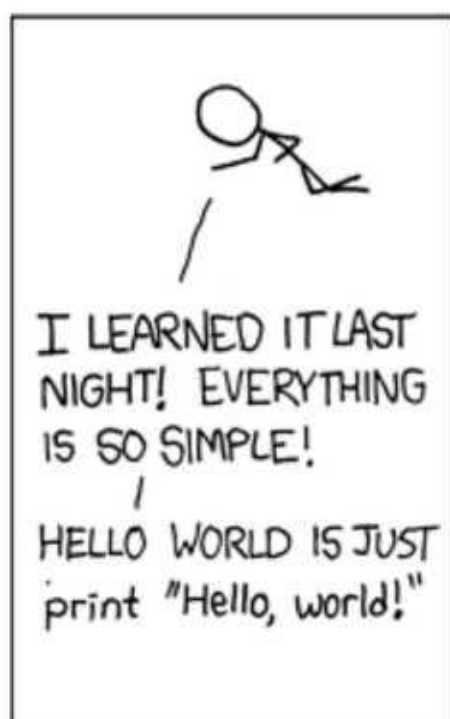
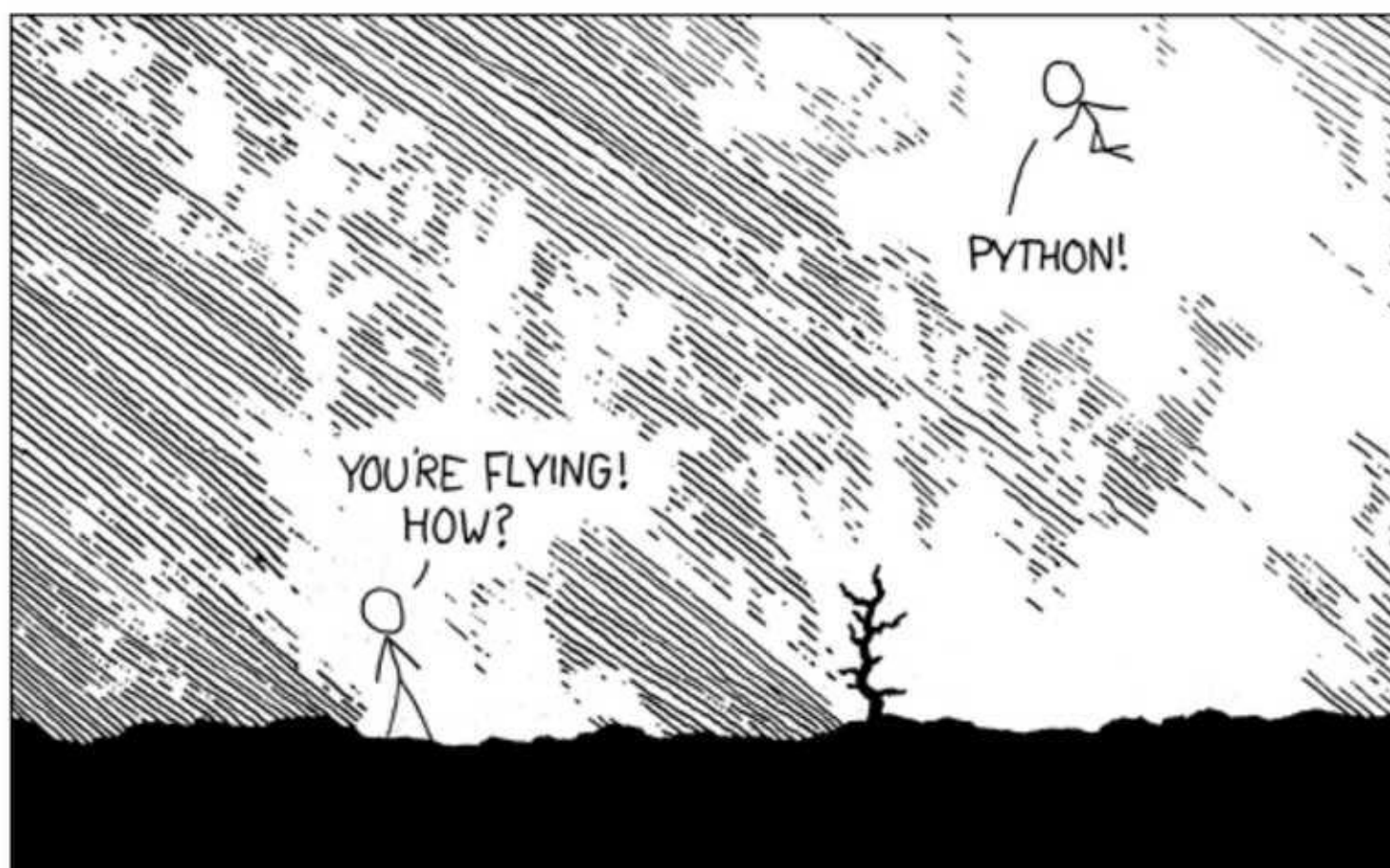
Python 追求的是找到最好的解决方案。相比之下，其他语言追求的是多种解决方案。

如果你试着读一段写的不错的 Python 代码，会发现像是在读英语一样。这也是 Python 的最大优点，它使你能够专注于解决问题而不是去搞明白语言本身。



注：漫画《口渴的 Python 开发者》，形容了 Python 开发者是多么轻松，来自 Pycot 网站。

其次，Python 功能强大，很多你本来应该操心的事情，Python 都替你考虑到了。当你用 Python 语言编写程序的时候，你不需要考虑如何管理你的程序使用的内存之类的底层细节。并且，Python 有很丰富的库，其中有官方的，也有第三方开发的，你想做的功能模块很有可能已经有人写好了，你只需要调用，不需要重新发明轮子。这就像是拥有了智能手机，可以任意安装需要的 app。



注：漫画 Python，作者xkcd

这幅漫画形容了 Python 的库有多强大，导入一个反重力库就可以飞起来了。

第三，Python 能做的事情有许多。

在职场中，使用 Python 工作的主要是这样几类人：

- 网站后端程序员：使用 Python 搭建网站、后台服务会比较容易维护，当需要增加新功能，用 Python 可以比较容易的实现。不少知名网站都使用了 Python 开发，比如：



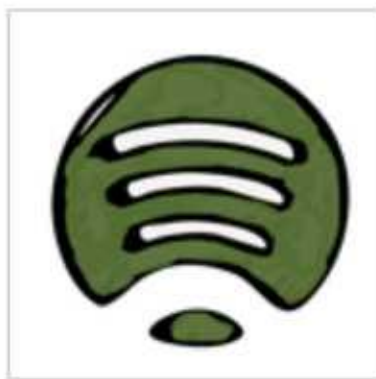
Gmail



Youtube



Reclit



Spotify



知乎



豆瓣

- 自动化运维：越来越多的运维开始倾向于自动化，批量处理大量的运维任务。Python 在系统管理上的优势在于强大的开发能力和完整的工具链。
- 数据分析师：Python 能快速开发的特性可以让你迅速验证你的想法，而不是把时间浪费在程序本身上，并且有丰富的第三方库的支持，也能帮你节省时间。
- 游戏开发者：一般是作为游戏脚本内嵌在游戏中，这样做的好处是即可以利用游戏引擎的高性能，又可以受益于脚本化开发的优点。只需要修改脚本内容就可以调整游戏内容，不需要重新编译游戏，特别方便。
- 自动化测试：对于测试来说，要掌握 Script 的特性，会在设计脚本中，有更好的效果。Python 是目前比较流行的 Script。

欢迎访问：电子书学习和下载网站 (<https://www.shgis.com>)
文档名称：《编程小白的第一本+python+入门书》侯爵 著.pdf
请登录 <https://shgis.com/post/4074.html> 下载完整文档。
手机端请扫码查看：

