作者：dylanninin  
链接：https://www.zhihu.com/question/20702054/answer/15960991  
来源：知乎  
著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。

入门和基础

Python有Python 2.x 和Python 3.x 之分，争论很多，见[Python2orPython3](https://link.zhihu.com/?target=https://wiki.python.org/moin/Python2orPython3" \t "_blank)。初学者不用考虑这个问题，可以从Python 2.x入门，之后再讨论会多一些理性。

[洪强宁：Python于Web 2.0网站的应用](https://link.zhihu.com/?target=http://www.slideshare.net/hongqn/qcon2010-3881323" \t "_blank)：豆瓣网洪强宁在QCon北京2010中的技术分享，若不能访问，请自备梯子；另外，豆瓣的[阿北](https://link.zhihu.com/?target=http://www.douban.com/people/ahbei/" \t "_blank)很值得关注。

[A Byte of Python](https://link.zhihu.com/?target=http://swaroopch.com/notes/python/" \t "_blank)：即简明Python教程，边看边练习就对Python有基本的了解，可以轻松存活。

[Google's Python Lessons](https://link.zhihu.com/?target=http://blog.hartleybrody.com/google-python/" \t "_blank)：Google出品的Python教程，值得信赖。

[Python Documentation](https://link.zhihu.com/?target=http://www.python.org/doc/" \t "_blank)：Python在线文档，若嫌枯燥，可以直接看[Python Standard Lib](https://link.zhihu.com/?target=http://docs.python.org/2/library/" \t "_blank)。

[42区：python入门指引](https://link.zhihu.com/?target=http://matrix.42qu.com/10757179" \t "_blank)：江湖人称[张教主](https://link.zhihu.com/?target=http://zuroc.42qu.com/" \t "_blank)的Python入门指引，除开[《Python核心编程》](https://link.zhihu.com/?target=http://book.douban.com/subject/3112503/" \t "_blank)。严格来说，这是张教主的一份Python资料索引。

[CS61A: SICP with Python](https://link.zhihu.com/?target=http://www-inst.eecs.berkeley.edu/~cs61a/fa11/61a-python/content/www/index.html" \t "_blank)：作为计算机相关人士，[SICP](https://link.zhihu.com/?target=http://en.wikipedia.org/wiki/Structure_and_Interpretation_of_Computer_Programs" \t "_blank)都不了解，实在很惭愧，所以来还债了。这应该是起源于[MIT的SICP](https://link.zhihu.com/?target=http://mitpress.mit.edu/sicp/" \t "_blank)教程，一个用Python，一个用Scheme。

[看到一个有趣的python的招聘测试](https://link.zhihu.com/?target=http://www.douban.com/group/topic/28872729/" \t "_blank)：同学发的一个贴，总结了一下应聘中遇到的问题，并给出了相应的资料来学习。如果想检测下自己的Python水平，请自觉移步[北京视讯天下的开发测试](https://link.zhihu.com/?target=http://www.video-tx.com/devtest/test.html%23/home" \t "_blank)。

MOOC们：在线公开课很多，见[课程图谱](https://link.zhihu.com/?target=http://coursegraph.com/search_results/python" \t "_blank)；仅在[Codecademy](https://link.zhihu.com/?target=http://www.codecademy.com/tracks/python" \t "_blank)上学了一点；现在希望能够每天跟进一些感兴趣的课程，多多学习。

书：去年6月份一同事离职，我买了两本[黑客与画家](https://link.zhihu.com/?target=http://book.douban.com/subject/6021440/" \t "_blank)，一本送给他(博客[一起去看海](https://link.zhihu.com/?target=http://www.ooobj.com/" \t "_blank))，另一本给部门老大；他回送了一本[Python源代码剖析](https://link.zhihu.com/?target=http://book.douban.com/subject/3117898/" \t "_blank)，看书名就知道是讲底层原理的，有机会啃啃。

源代码：直接到[Github](https://link.zhihu.com/?target=https://github.com/search?q=python&ref=cmdform" \t "_blank)上去找吧。

Python与数据库

在项目开发中，数据库应用必不可少。这里汇总下目前接触过的数据库和使用教程。

SQL，计算机出身的人应该都学过，若有疑问可以参考[Wikipedia SQL](https://link.zhihu.com/?target=http://en.wikipedia.org/wiki/SQL" \t "_blank)，并使用SQLite做下练习。

[PEP249: DB API](https://link.zhihu.com/?target=http://www.python.org/dev/peps/pep-0249/" \t "_blank)：数据库访问接口规范，当时还做了一份[笔记](https://link.zhihu.com/?target=http://dylanninin.com/blog/2012/11/16/python_pep249.html" \t "_blank)。

[SQLite](https://link.zhihu.com/?target=http://www.sqlite.org/" \t "_blank)：教程见[SQLite Python Tutorial](https://link.zhihu.com/?target=http://www.tutorialspoint.com/sqlite/sqlite_python.htm" \t "_blank)；若要深入了解，推荐[The Definitive Guide to SQLite](https://link.zhihu.com/?target=http://book.douban.com/subject/5392299/" \t "_blank)；看此书时，做了一些好句子、段落的摘要，见[Sentences in SQLite 3](https://link.zhihu.com/?target=http://dylanninin.com/blog/2013/11/10/sentences_in_sqlite.html" \t "_blank)。

[MySQL](https://link.zhihu.com/?target=http://www.mysql.com/" \t "_blank)：教程见[Python MySQL Database Access](https://link.zhihu.com/?target=http://www.tutorialspoint.com/python/python_database_access.htm" \t "_blank)。

[Oracle Database](https://link.zhihu.com/?target=http://www.oracle.com/us/products/database/overview/index.html" \t "_blank)：有可能这是最庞大的数据库，所以需要专职的DBA。官方教程[The Mastering Oracle+Python Series](https://link.zhihu.com/?target=http://www.oracle.com/technetwork/articles/dsl/mastering-oracle-python-1391323.html" \t "_blank)，快速入门教程[cx\_Oracle Quick Start](https://link.zhihu.com/?target=http://dbaportal.eu/sidekicks/sidekick-cx_oracle-code-paterns/" \t "_blank)。

[SQLAlchemy](https://link.zhihu.com/?target=http://www.sqlalchemy.org/" \t "_blank)：Python的ORM标准和框架，解决面向对象编程和关系数据库模式不匹配的问题。

数据库理论除了SQL，还有另一派NoSQL。

关于常见NoSQL产品的比较见[Main NoSQL Database Comparison](https://link.zhihu.com/?target=http://kkovacs.eu/cassandra-vs-mongodb-vs-couchdb-vs-redis" \t "_blank)；若果需要一些NoSQL的理论知识和基本概念，见[The NoSQL Ecosystem](https://link.zhihu.com/?target=http://www.aosabook.org/en/nosql.html" \t "_blank)、[Big Data与NoSQL](https://link.zhihu.com/?target=http://www.slideshare.net/pavlobaron/big-data-nosql-efs11-pavlo-baron" \t "_blank)。目前仅用过[MongoDB](https://link.zhihu.com/?target=http://www.mongodb.org/" \t "_blank)和[Redis](https://link.zhihu.com/?target=http://redis.io/" \t "_blank)。

MongoDB：如果熟悉SQL，MongDB的学习成本会很低；相关资料见[Python Language Center in MongoDB](https://link.zhihu.com/?target=http://docs.mongodb.org/ecosystem/drivers/python/" \t "_blank)；用过的两个驱动：1）[PyMongo](https://link.zhihu.com/?target=http://api.mongodb.org/python/current/" \t "_blank)，提供了类似Mongo Shell的接口；2）[MongoEngine](https://link.zhihu.com/?target=http://mongoengine.org/" \t "_blank): A Python Object-Document-Mapper for working with MongoDB，即MongoDB的'ORM'框架，此时变成了'ODM'，[MongoEngine on Github](https://link.zhihu.com/?target=https://github.com/hmarr/mongoengine" \t "_blank)。

Redis：Redis需要一些学习成本，入门推荐[The Little Redis Book](https://link.zhihu.com/?target=http://openmymind.net/2012/1/23/The-Little-Redis-Book/" \t "_blank)；用过的Python客户端驱动[Redis-py](https://link.zhihu.com/?target=https://github.com/andymccurdy/redis-py" \t "_blank)；更多客户端见[Redis Clients](https://link.zhihu.com/?target=http://redis.io/clients" \t "_blank)。

NoSQL建模：SQL发展了几十年，有很成熟的建模技术，那么NoSQL呢，见[陈皓：NoSQL数据建模技术](https://link.zhihu.com/?target=http://coolshell.cn/articles/7270.html" \t "_blank)，原文[NoSQL Data Modeling Techniques](https://link.zhihu.com/?target=http://highlyscalable.wordpress.com/2012/03/01/nosql-data-modeling-techniques/" \t "_blank)。

更多资料：NoSQL英文站点见[NoSQL Database](https://link.zhihu.com/?target=http://nosql-database.org/" \t "_blank)；NoSQL中文论坛见[NoSQL Fan：关注NoSQl相关的新闻和技术](https://link.zhihu.com/?target=http://blog.nosqlfan.com/" \t "_blank)。NoSQL Fan中，MongoDB和Redis资料很多，已经形成了资料专题，包括介绍、内部实现、应用与优化、新闻等，总能发现你想要的东西：1）[NoSQL Fan：Redis资料汇总专题](https://link.zhihu.com/?target=http://blog.nosqlfan.com/html/3537.html" \t "_blank)；2）[NoSQL Fan: MongoDB资料汇总专题](https://link.zhihu.com/?target=http://blog.nosqlfan.com/html/3548.html" \t "_blank)。

Python与Web开发

Python的Web框架众多，见[Web Frameworks for Python](https://link.zhihu.com/?target=https://wiki.python.org/moin/WebFrameworks" \t "_blank)，总有一款适合你或你的项目，实在不行，请动手打造自己的框架；为什么会有这么多框架呢，见[Why so many Python wen frameworks?](https://link.zhihu.com/?target=http://bitworking.org/news/Why_so_many_Python_web_frameworks" \t "_blank)。

[Web.py](https://link.zhihu.com/?target=http://webpy.org/" \t "_blank)：已故[Aaron Swartz](https://link.zhihu.com/?target=http://www.aaronsw.com/" \t "_blank)的框架。一句话介绍"web.py is a web framework for Python that is as simple as it is powerful. web.py is in the public domain; you can use it for whatever purpose with absolutely no restrictions."。接触的第一个Web框架，后来模仿MovableType，写了一个简单的博客，见[Blog on Github](https://link.zhihu.com/?target=https://github.com/dylanninin/blog/" \t "_blank)，在线demo [Pastime Paradise](https://link.zhihu.com/?target=http://ec2-54-251-227-120.ap-southeast-1.compute.amazonaws.com/" \t "_blank)；碰到的坑点是模板中嵌套Python代码一直有缩进问题。

[Flask](https://link.zhihu.com/?target=http://flask.pocoo.org/" \t "_blank)：一句话介绍"Flask is a lightweight web application framework written in Python and based on the Werkzeug WSGI toolkit and Jinja2 template engine. It is BSD licensed. Flask is called a microframework because it keeps the core simple but extensible"。因为工作变动，练习过一段时间的Flask，以便熟悉开发的工具链；如果没有Rails，我想这才是自己首选的Web开发框架: simple but extensible and for fun。实战教程[The Flask Mega-Tutorial](https://link.zhihu.com/?target=http://blog.miguelgrinberg.com/post/the-flask-mega-tutorial-part-i-hello-world" \t "_blank)。

[Tornado](https://link.zhihu.com/?target=http://www.tornadoweb.org/en/stable/" \t "_blank):来自[FriendFeed](https://link.zhihu.com/?target=http://friendfeed.com/" \t "_blank)的异步框架，FriendFeed被Facebook收购后开放了源代码，见[Tornado on Github](https://link.zhihu.com/?target=https://github.com/facebook/tornado" \t "_blank)。一句话介绍"Tornado is a Python web framework and asynchronous networking library, originally developed at FriendFeed"。Tornado主要特点是[non-blocking](https://link.zhihu.com/?target=http://en.wikipedia.org/wiki/Non-blocking_I/O" \t "_blank)，如果你想开发[real-time](https://link.zhihu.com/?target=http://en.wikipedia.org/wiki/Real-time_web" \t "_blank)的Web应用，Tornado是一个不错的选择。

[Django](https://link.zhihu.com/?target=https://www.djangoproject.com/" \t "_blank)：一句话介绍"Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design."，接触不多，作为全栈式框架，听说它的组件都是Made in Django。更多资料见[Django资料](https://link.zhihu.com/?target=http://haoluobo.com/trac/wiki/Django" \t "_blank)。

IDE

* [Vim](https://link.zhihu.com/?target=http://www.vim.org/" \t "_blank) + [Python Mode](https://link.zhihu.com/?target=https://github.com/klen/python-mode" \t "_blank)。目前就用这个方案，很方便，直接引用同学的[效果图](https://link.zhihu.com/?target=http://www.douban.com/photos/photo/1975213817/" \t "_blank)。关于Vim资料和讨论，请移步[Vim资料大全](https://link.zhihu.com/?target=http://wiki.hotoo.me/Vim.html" \t "_blank)；关于Vim更多插件和演示，请移步[k-vim on Github](https://link.zhihu.com/?target=https://github.com/wklken/k-vim" \t "_blank)；关于Git，Github，请移步本站[Free Blog with Github Pages](https://link.zhihu.com/?target=http://dylanninin.com/blog/2013/11/02/free_blogs.html" \t "_blank)。

社区

* [啄木鸟社区](https://link.zhihu.com/?target=http://wiki.woodpecker.org.cn/moin/" \t "_blank)
* [42区：网站开发.漫游指南](https://link.zhihu.com/?target=http://matrix.42qu.com/" \t "_blank)
* [Python on V2EX](https://link.zhihu.com/?target=https://www.v2ex.com/go/python" \t "_blank)
* [Python4cn](https://link.zhihu.com/?target=http://www.simple-is-better.com/" \t "_blank)
* [豆瓣Python小组](https://link.zhihu.com/?target=http://www.douban.com/group/python/" \t "_blank)
* [哲思社区](https://link.zhihu.com/?target=http://www.zeuux.com/" \t "_blank)

周刊

* [Python Weekly](https://link.zhihu.com/?target=http://www.pythonweekly.com/" \t "_blank)：每周更新，包括Python相关的文章、教程、演讲、书籍、项目、工作等。
* [Pycoder's Weekly](https://link.zhihu.com/?target=http://www.pycoders.com/" \t "_blank)：与Python Weekly类似，两者可以互为补充，了解过去一周动态。
* [码农周刊](https://link.zhihu.com/?target=http://weekly.manong.io/" \t "_blank): [developerWorks](https://link.zhihu.com/?target=http://weibo.com/developerworks" \t "_blank)出品的周刊，来自国人的分享。可以先看[为什么要做《码农周刊》？](https://link.zhihu.com/?target=http://blog.manong.io/why-create-a-weekly-newsletter-for-programmers/" \t "_blank)；接着[《码农周刊》用到的一些技术](https://link.zhihu.com/?target=http://blog.manong.io/technologies-we-use/" \t "_blank)； 再接着[周刊回顾](https://link.zhihu.com/?target=http://weekly.manong.io/issues/" \t "_blank)。这里不仅仅是Python。

更新 2016-03-06  
Python 3

前面提到，『Python有Python 2.x 和Python 3.x 之分，争论很多，见[Python2orPython3](https://link.zhihu.com/?target=https://wiki.python.org/moin/Python2orPython3" \t "_blank)。初学者不用考虑这个问题，可以从Python 2.x入门，之后再讨论会多一些理性。』目前来看，已然不妥：对于新手，推荐直接学习 Python 3.x；对于还停留在 Python 2.x 的，则应尽快拥抱变化，与时俱进。

对于项目的开发维护来说，在新旧版本中来回穿梭/切换是一件痛苦的事儿，也会增加很多不必要的负担，因此尽快推进将手头的项目升级到 Python 3.x（当前最新版本为 3.5）。但升级绝不是拍拍脑袋就能决定的，具体参考 [Porting Python 2 Code to Python 3](https://link.zhihu.com/?target=https://docs.python.org/dev/howto/pyporting.html" \t "_blank)。在 2015 年年底，便将一个开发维护近半年的 Web 项目顺利从 2.7 升级到 3.5，这与单元测试覆盖足够、第三方依赖库都支持 Python 3、团队成员愿意持续学习是分不开的。

* [Key Differences between 2.7.x and 3.x](https://link.zhihu.com/?target=http://sebastianraschka.com/Articles/2014_python_2_3_key_diff.html" \t "_blank)
* [Writing Python 2-3 compatible code](https://link.zhihu.com/?target=http://python-future.org/compatible_idioms.html" \t "_blank)
* [Porting Python 2 Code to Python 3](https://link.zhihu.com/?target=https://docs.python.org/dev/howto/pyporting.html" \t "_blank)
* [Python 3.5: What’s new](https://link.zhihu.com/?target=https://docs.python.org/3/whatsnew/3.5.html" \t "_blank)
* [Python 3 Glossory](https://link.zhihu.com/?target=https://docs.python.org/3/glossary.html" \t "_blank)
* [Python 3 Cookbook on Github](https://link.zhihu.com/?target=https://github.com/yidao620c/python3-cookbook" \t "_blank)