

知乎问题回答

Python学习完基础语法知识后，如何进一步提高？

如果你已经完成了Python基础语法的学习，想要知道接下来如何提高，那么你得先问问自己你要用Python来做什么？目前学习Python后可能的就业方向包括以下几个领域，我把每个领域需要的技术作为了一个简单的关键词摘要。

说明：以下数据参考了主要的招聘门户网站以及职友集。

职位	所需技能	招聘需求量
Python后端开发工程师	Python基础 Django / Flask / Tornado / Sanic RESTful / 接口文档撰写 MySQL / Redis / MongoDB / ElasticSearch Linux / Git / Scrum / PyCharm	大
Python爬虫开发工程师	Python基础 常用标准库和三方库 Scrapy / PySpider Selenium / Appnium Redis / MongoDB / MySQL 前端 / HTTP(S) / 抓包工具	较少
Python量化交易开发工程师	Python基础 数据结构 / 算法 / 设计模式 NoSQL (KV数据库) 金融学 (两融、期权、期货、股票) / 数字货币	较大 (一线城市)
Python数据分析工程师 / Python机器学习工程师	统计学专业 / 数学专业 / 计算机专业 Python基础 / 算法设计 SQL / NoSQL / Hive / Hadoop / Spark NumPy / Scikit-Learn / Pandas / Seaborn PyTorch / Tensorflow / OpenCV	较大 (一线城市)
Python自动化测试工程师	Python基础 / 单元测试 / 软件测试基础 Linux / Shell / JIRA / 禅道 / Jenkins / CI / CD Selenium / Robot Framework / Appnium ab / sysbench / JMeter / LoadRunner / QTP	大
Python自动化运维工程师	Python基础 / Linux / Shell Fabric / Ansible / Playbook Zabbix / Saltstack / Puppet Docker / paramiko	较大 (一线城市)
Python云平台开发工程师	Python基础 OpenStack / CloudStack Ovirt / KVM Docker / K8S	较少 (一线城市)

如果弄清了自己将来要做的方向，就可以开始有针对性的学习了，下面给大家一个推荐书籍的清单。

1. 入门读物

- 《Python基础教程》 (*Beginning Python From Novice to Professional*)
- 《Python学习手册》 (*Learning Python*)
- 《Python编程》 (*Programming Python*)
- 《Python编程从入门到实践》 (*Python Crash Course*)
- 《Python Cookbook》

2. 进阶读物

- 《软件架构 - Python语言实现》 (*Software Architecture with Python*)
- 《流畅的Python》 (*Fluent Python*)
- 《Python设计模式》 (*Learning Python Design Patterns*)
- 《Python高级编程》 (*Expert Python Programming*)
- 《Python性能分析与优化》 (*Mastering Python High Performance*)

3. 数据库相关

- 《MySQL必知必会》 (*MySQL Crash Course*)
- 《深入浅出MySQL - 数据库开发、优化与管理维护》
- 《MongoDB权威指南》 (*MongoDB: The Definitive Guide*)
- 《Redis实战》 (*Redis in Action*)
- 《Redis开发与运维》

4. Linux / Shell / Docker / 运维

- 《鸟哥的Linux私房菜》
- 《Linux命令行与shell脚本编程大全》 (*Linux Command Line and Shell Scripting Bible*)
- 《Python自动化运维:技术与最佳实践》
- 《第一本Docker书》 (*The Docker Book*)
- 《Docker经典实例》 (*Docker Cookbook*)

5. Django / Flask / Tornado

- 《Django基础教程》 (*Tango with Django*)
- 《轻量级Django》 (*Lightweight Django*)
- 《精通Django》 (*Mastering Django: Core*)
- 《Python Web开发：测试驱动方法》 (*Test-Driven Development with Python*)
- 《Two Scoops of Django: Best Practice of Django 1.8》
- 《Flask Web开发：基于Python的Web应用开发实战》 (*Flask Web Development: Developing Web Applications with Python*)
- 《深入理解Flask》 (*Mastering Flask*)
- 《Introduction to Tornado》

6. 爬虫开发

- 《用Python写网络爬虫》 (*Web Scraping with Python*)
- 《精通Python爬虫框架Scrapy》 (*Learning Scrapy*)
- 《Python网络数据采集》 (*Web Scraping with Python*)
- 《Python爬虫开发与项目实战》
- 《Python 3网络爬虫开发实战》

7. 数据分析

- 《利用Python进行数据分析》 (*Python for Data Analysis*)
- 《Python数据科学手册》 (*Python Data Science Handbook*)
- 《Python金融大数据分析》 (*Python for Finance*)
- 《Python数据可视化编程实战》 (*Python Data Visualization Cookbook*)
- 《Python数据处理》 (*Data Wrangling with Python*)

8. 机器学习

- 《Python机器学习基础教程》 (*Introduction to Machine Learning with Python*)

- 《Python机器学习实践指南》 (*Python Machine Learning Blueprints*)
- 《Python机器学习实践：测试驱动的开发方法》 (*Thoughtful Machine Learning with Python A Test Driven Approach*)
- 《Python机器学习经典实例》 (*Python Machine Learning Cookbook*)
- 《TensorFlow：实战Google深度学习框架》

9. 其他书籍

- 《Pro Git》
- 《Selenium自动化测试 - 基于Python语言》 (*Learning Selenium Testing Tools with Python*)
- 《Selenium自动化测试之道》
- 《Scrum敏捷软件开发》 (*Software Development using Scrum*)
- 《高效团队开发 - 工具与方法》

当然学习编程，最重要的通过项目实战来提升自己的综合能力，Github上有大量的优质开源项目，其中不乏优质的Python项目。有一个名为[“awesome-python-applications”](https://github.com/awesome-python/awesome-python-applications)的项目对这些优质的资源进行了归类并提供了传送门，大家可以了解下。如果自学能力不是那么强，可以通过网络上免费或者付费的视频课程来学习对应的知识；如果自律性没有那么强，那就只能建议花钱参加培训班了，因为花钱在有人监督的环境下学习对很多人来说确实是一个捷径，但是要记得：“师傅领进门，修行靠各人”。选择自己热爱的东西并全力以赴，不要盲目的跟风学习，这一点算是过来人的忠告吧。记得我自己刚开始进入软件开发这个行业时，有人跟我说过这么一句话，现在也分享出来与诸君共勉：“浮躁的人有两种：只观望而不学习的人，只学习而不坚持的人；浮躁的人都不是高手。”