

目录

第 1 章 入门准备

01 开篇词：你为什么要学 Python ？

02 我会怎样带你学 Python ？

03 让 Python 在你的电脑上安家落户

04 如何运行 Python 代码 ？

第 2 章 通用语言特性

05 数据的名字和种类—变量和类型

06 一串数据怎么存—列表和字符串

07 不只有一条路—分支和循环

08 将代码放进盒子—函数

09 知错能改—错误处理、异常机制

10 定制一个模子—类

11 更大的代码盒子—模块和包

12 练习—密码生成器

第 3 章 Python 进阶语言特性

13 这么多的数据结构（一）：列表、元祖、字符串

14 这么多的数据结构（二）：字典、集合

15 Python大法初体验：内置函数

16 深入理解下迭代器和生成器

17 生成器表达式和列表生成式

18 把盒子升级为豪宅：函数进阶

19 让你的模子更好用：类进阶

20 从小独栋升级为别墅区：函数式编程

## 27 这么多条路，你想走那一条？

更新时间：2019-10-28 10:17:03



“立志是事业的大门，工作是登门入室的旅途。”  
——巴斯德

到此为止，Python 基础知识的学习告一段落，接下来，该如何进一步提高呢？

### 最重要的是练习

学习编程，不能简单地依靠阅读来提升编程能力，读再多的文章，看再多的视频，如果不实际动手去练习实践，是无法真正掌握编程的。

好比《天龙八部》中的段誉，偶然习得六脉神剑，武功对决中却时灵时不灵，无法自由施展，这便是练习不足所致。后来段誉多次运用六脉神剑后，逐渐积累经验，领悟施展方法，方才渐入佳境。

练习，一方面强化了你对已学编程知识的记忆，增进理解。另一方面，让你在实践中掌握编程思维。

编程思维是一种特别的思维方式，将一个抽象笼统的需求转化为能用程序解决的问题，它不存在于生活中，只存在于代码中，只能靠不断的练习来领悟和强化。

所以，学习编程的重中之重是，练习。不做练习学习再多也是纸上谈兵。

练习从哪种题目（项目）开始呢？

对于编程初学者来讲，功能简单的练习尤佳，尽量不涉及复杂知识，能用十几行、几十行代码来解决一个问题就足够了。

得益于开源，我们能在 Github 中轻易找到一些有实用价值和趣味的 Python 入门级练习题目。比如：<https://github.com/geekcomputers/Python>。

<div>← 慕课专栏</div> <div>三 你的第一本Python基础入门书 / 27 这么多条路，你想走那一条？</div>	
目录	
第 1 章 入门准备	
01 开篇词：你为什么要学 Python ？	
02 我会怎样带你学 Python ？	
03 让 Python 在你的电脑上安家落户	
04 如何运行 Python 代码？	<ul style="list-style-type: none"><li><code>base_dir.py</code> - This script returns a group of files in a given directory, once you pass the current and the new extensions.</li><li><code>create_dir_if_not_there.py</code> - Checks to see if a directory exists in the users home directory, if not then create it.</li><li><code>Fast Youtube Downloader</code> - Downloads YouTube videos quickly with parallel threads using aria2c</li><li><code>Google Image Downloader</code> - Query a given term and retrieve images from the Google Image database.</li><li><code>dir_test.py</code> - Tests to see if the directory <code>testdir</code> exists, if not it will create the directory for you.</li><li><code>env_check.py</code> - This script will check to see if all of the environment variables required are set.</li><li><code>blackjack.py</code> - This script contains the Casino BlackJack-21 Game in Python.</li><li><code>fileinfo.py</code> - Shows file information for a given file.</li><li><code>folder_size.py</code> - Scans the current directory and all subdirectories and displays the size.</li><li><code>logs.py</code> - This script will search for all <code>*.log</code> files in the given directory, zip them using the program you specify, and then date stamp them.</li><li><code>move_files_over_x_days.py</code> - Moves all files over a specified age (in days) from the source directory to the destination directory.</li><li><code>nslookup_check.py</code> - This simple script opens the file <code>server_list.txt</code> and then does an nslookup for each one to check the DNS entry.</li><li><code>osinfo.py</code> - Displays some information about the OS on which you are running this script.</li><li><code>ping_servers.py</code> - This script, depending on the arguments supplied, will ping the servers associated with that application group.</li></ul>
第 2 章 通用语言特性	
05 数据的名字和种类—变量和类型	
06 一串数据怎么存—列表和字符串	
07 不只有一条路—分支和循环	
08 将代码放进盒子—函数	
09 知错能改—错误处理、异常机制	
10 定制一个模子—类	
11 更大的代码盒子—模块和包	
12 练习—密码生成器	
第 3 章 Python 进阶语言特性	
13 这么多的数据结构（一）：列表、元祖、字符串	
14 这么多的数据结构（二）：字典、集合	
15 Python大法初体验：内置函数	
16 深入理解下迭代器和生成器	
17 生成器表达式和列表生成式	
18 把盒子升级为豪宅：函数进阶	<ul style="list-style-type: none"><li><code>datetime</code>：日期和时间处理相关</li><li><code>random</code>：随机取值相关</li><li><code>json</code>：json 相关</li><li><code>XML</code>：XML 相关</li><li><code>collections</code>：集合相关</li><li><code>base64</code>：base64 编码相关</li><li><code>hashlib</code>：摘要算法相关，如 MD5、SHA1</li><li><code>itertools</code>：迭代工具相关</li><li><code>contextlib</code>：上下文管理相关</li><li><code>urllib</code>：HTTP 请求相关</li></ul>
19 让你的模子更好用：类进阶	
20 从小独栋升级为别墅区：函数式编	

例如上面有，文件批量重命名、创建目录、测试文件是否存在、查看文件信息、检查目录大小等等练习，涉及的知识也很简单，非常适合大家练习。加油吧！

学习标准库

Python 标准库是 Python 的一大利器。有非常多的常用功能被 Python 社区开发好并置入标准库中，对于开发者而言，只要拿来使用即可，大大节省开发时间。

其中常用的库有：

- `datetime`：日期和时间处理相关
- `random`：随机取值相关
- `json`：json 相关
- `XML`：XML 相关
- `collections`：集合相关
- `base64`：base64 编码相关
- `hashlib`：摘要算法相关，如 MD5、SHA1
- `itertools`：迭代工具相关
- `contextlib`：上下文管理相关
- `urllib`：HTTP 请求相关

标准库就像一个工具箱，里面有很多功能，我们不可能一下子全部掌握其用法，但是我们可以先看看工具箱中有什么工具，也就是了解其中有哪些功能，在需要某个功能时能立即联想到这个工具箱中的工具，然后查阅说明书（文档）来使用。

<div>← 慕课专栏</div> <div>≡ 你的第一本Python基础入门书 / 27 这么多条路，你想走那一条？</div>	
目录	当然，标准库也不是万能，它只提供基础的、常用的功能，如果你想要的功能不在标准库中，可以尝试引入第三方包来解决。可以在 <a href="#">Python Package Index</a> 中查找所需要的包。
第 1 章 入门准备	选择具体方向
01 开篇词：你为什么要学 Python ？	学习了 Python 基础知识后，完全可以用 Python 来编写程序了，不过这只限于一些简单的程序，可能只够满足自己的一些日常需求。要想编写更大型的程序，比如一个网站，所涉及的知识就复杂不少了，已经不能只靠编程语言和简单的库来解决了，因为编程语言只是计算机世界中的一个小的领域，并不能解决一切问题。
02 我会怎样带你学 Python ？	沿着 Python 这条路线继续向后走，还有很多的方向可选：
03 让 Python 在你的电脑上安家落户	<div><div>• Web 开发</div><div>• 数据科学</div><div>• 爬虫系统</div><div>• 机器学习</div><div>• 自动化运维和测试</div><div>• .....</div></div>
04 如何运行 Python 代码？	每个方向会都涉及很多不同的知识，比如 Web 开发可能涉及网络、数据库、缓存等，自动化运维可能涉及容器、持续集成、云等。
第 2 章 通用语言特性	对于这些方向，我们在下一章中具体来看看。
05 数据的名字和种类—变量和类型	<div>← 26 更加 Python 的 Python 代码风格</div> <div>28 Web 开发 →</div>
06 一串数据怎么存—列表和字符串	
07 不只有一条路—分支和循环	
08 将代码放进盒子—函数	
09 知错能改—错误处理、异常机制	
10 定制一个模子—类	
11 更大的代码盒子—模块和包	
12 练习—密码生成器	
第 3 章 Python 进阶语言特性	
13 这么多的数据结构（一）：列表、元祖、字符串	精选留言 0
14 这么多的数据结构（二）：字典、集合	欢迎在这里发表留言，作者筛选后可公开显示
15 Python大法初体验：内置函数	
16 深入理解下迭代器和生成器	
17 生成器表达式和列表生成式	
18 把盒子升级为豪宅：函数进阶	
19 让你的模子更好用：类进阶	
20 从小独栋升级为别墅区：函数式编程	

## 目录

### 第 1 章 入门准备

01 开篇词：你为什么要学 Python ？

02 我会怎样带你学 Python ？

03 让 Python 在你的电脑上安家落户

04 如何运行 Python 代码 ？

### 第 2 章 通用语言特性

05 数据的名字和种类——变量和类型

06 一串数据怎么存——列表和字符串

07 不只有一条路——分支和循环

08 将代码放进盒子——函数

09 知错能改——错误处理、异常机制

10 定制一个模子——类

11 更大的代码盒子——模块和包

12 练习——密码生成器

### 第 3 章 Python 进阶语言特性

13 这么多的数据结构（一）：列表、元祖、字符串

14 这么多的数据结构（二）：字典、集合

15 Python大法初体验：内置函数

16 深入理解下迭代器和生成器

17 生成器表达式和列表生成式

18 把盒子升级为豪宅：函数进阶

19 让你的模子更好用：类进阶

20 从小独栋升级为别墅区：函数式编程