

目录

第 1 章 入门准备

01 开篇词：你为什么要学 Python [最近阅读](#)

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

01 开篇词：你为什么要学 Python？

更新时间：2019-10-30 13:47:01



“今天应做的事没有做，明天再早也是耽误了。”  
——裴斯泰洛齐

你好，欢迎阅读本专栏内容。我是讲师黄浮云，云计算资深工程师，曾深入使用多种程序语言，其中便包括本专栏中为大家讲解的 Python，希望这个专栏在帮助读者顺利掌握 Python 的同时，也能让读者领悟学习任何一门语言的高效方法。

当你打开了这篇文章，我知道你一定是个饱含学习热情的人。现在你可能对 Python 有一些想法，想要让它为你所用，但同时可能有一丝犹豫。你不知道这个专栏的深度适不适合你。放心，无论你是否曾经是否有编程经验，学习 Python 这个专栏都是合适你的。另外你可能会想，Python 是不是一个好的选择？

Python 是不是一个好选择

如果你从未有过编程经验，无论你是想将编程作为业余爱好，还是只想体验编程思维，或者是想以此为开端今后从事软件开发工作，从 Python 入手学习编程都是个不错的选择。

对于新手而言 Python 的优势有：

- 1. 上手容易
- 2. 第三方库众多无所不包
- 3. 相较于 C / C++ / Java 等语言有更强的表现力，这也意味着实现相同的功能需要的代码量更少
- 4. 上面的 2. 和 3. 可以从侧面说明 Python 有着很高的开发效率
- 5. 和其他解释型语言一样，可交互式地执行代码，便于快速试验

同样的，如果你有编程经验，并且想在软件开发这个领域继续深入发展，通过学习 Python 来拓展自己的技能和知识面也不失为一个好的选择。

← 慕课专栏	☰ 你的第一本Python基础入门书 / 01 开篇词：你为什么要学 Python？
目录	多努力，入错了门那沉没成本就太高了。其实大可不必如此。
第 1 章 入门准备	对于程序语言而言，很难断定究竟谁是最好的。历史上程序语言有很多，经过时间淘洗留下来的并且还在广泛使用的，自然有其价值。虽然就语言设计而言，各个程序语言间可能存在一些设计上的优劣，但每个语言都有其独有的优势和擅长的应用场景。
01 开篇词：你为什么要学 Python？ <a href="#">最近阅读</a>	想找出最好的那门语言其实也是没必要的，不能指望自己的整个职业生涯可以靠一门语言一招鲜吃遍天。程序员的职业生涯中会因为很多原因而需要学习新的语言，可能是因为你换了公司或部门，也可能你的所在的团队需要应对新的场景。
02 我会怎样带你学 Python？	每个程序员都应该主动去学习多门语言，去感受多种不同语言背后的思想，而 Python 是你应该去学习的语言之一。Python 可能是目前市面上使用场景最广泛的语言，普遍应用于数据科学，机器学习，Web 开发，爬虫，自动化运维，自动化测试等领域。并且 Python 也具备多样的编程范式，如过程式，面向对象，以及一些函数式特性。面对着这样一个多面手，即使眼前暂无需要，但在日后终究会有好处，更何况艺多不压身是吧
03 让 Python 在你的电脑上安家落户	Python未来潜力如何
04 如何运行 Python 代码？	如果我们从语言易用性，成熟度，社区规模，典型应用场景，以及衍生的一系列技术栈等角度来看，Python 的表现是很优秀的。近年来的热门领域，如机器学习，数据科学，Python 也有着举足轻重的地位。加之越来越多的国内外高校开始将 Python 作为编程入门教学语言，Python 的使用基数在持续扩大。在笔者看来，Python 在未来还会有一番天地。
第 2 章 通用语言特性	Python 2 or Python 3
05 数据的名字和种类—变量和类型	在过去的很长一段时间中，学习 Python 的人常常在 Python 2 和 Python 3 版本间陷入选择困难。因为这两个版本存在着较大的不兼容，而 Python 2 由于历史更加悠久所以积累的第三方库要比 Python 3 多很多，同时由于将代码由 Python 2 迁移到 Python 3 不是一件容易的事，所以在 Python 3 发布后几年时间□，大家并没有太大动力使用这个新的版本，直接导致 Python 2 和 Python 3 在过去几年间割裂得十分严重。
06 一串数据怎么存—列表和字符串	但是时间已经来到了 2019 年，大量的代码从 Python 2 版本迁移到了 Python 3，Python 官方也将在2020 年停止维护 Python 2 版本。所以 Python 2 和 Python 3 的选择已不再是个问题，读者朋友也不必再纠结怎么选择，请直接选 Python 3 吧，本专栏也将使用 Python 3 作为教学版本。
07 不只有一条路—分支和循环	开始 Python 之旅
08 将代码放进盒子—函数	好了，主意一旦拿定就算是扬起了帆，我们该起航了，Python 之旅正式开始。
09 知错能改—错误处理、异常机制	02 我会怎样带你学 Python？→
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	

← 慕课专栏	≡ 你的第一本Python基础入门书 / 01 开篇词：你为什么要学 Python ？
目录	很棒很棒 <div>👍 1    回复</div> <div>2019-11-27</div>
第 1 章 入门准备	
01 开篇词：你为什么要学 Python ？ <a href="#">最近阅读</a>	<b>慕九州22958251</b> 很棒啊 <div>👍 0    回复</div> <div>2019-11-27</div>
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 ？	<b>慕田峪1440454</b> 作为非计算机专业的准大一新生，非常希望能掌握这项新技能，却也感觉到了很多对小白不友好的门槛，（ 这是啥？那个又是啥？）迫切需要一位引路人拨开迷雾 希望能够在这里有收获！ <div>👍 5    回复</div> <div>2019-08-30</div>
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	<b>weixin_慕桂英6357015</b> 这课是文字阅读还是互动视频？ <div>👍 0    回复</div> <div>2019-08-27</div>
08 将代码放进盒子—函数	<b>DongHj</b> 回复 <b>weixin_慕桂英6357015</b> 同学你好，课程是文字专栏，不是视频课哦 回复 <div>2019-09-02 09:59:20</div>
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	<b>黄浮云</b> 我是讲师黄浮云，大家有任何问题可以在这里留言，也可以加我微信（floccloud），一起沟通讨论。 <div>👍 4    回复</div> <div>2019-08-22</div>
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	<b>不爱吃饺子</b> 回复 <b>黄浮云</b> 老师你好，学习python已经两三个月了，所以python基础的一些知识都已经了解的差不多了，您这门课对于像我这样的学生会有哪些帮助，您能说一下吗？ <div>回复</div> <div>2019-08-24 13:39:00</div>
14 这么多的数据结构（二）：字典、集合	<b>黄浮云</b> 回复 <b>不爱吃饺子</b> 我不了解你之前的学习到了怎样的程度，来做个自评吧。如果你了解生成器原理；能自定义一个装饰器；知道类中常见特殊方法的用途（诸如 __getitem__() ），那么你的 Python 基础知识应该还不错，你应该去学习更高阶的内容。如果上面问题有两个以上你不了解，那么这个课程应该会对对你有帮助。 <div>回复</div> <div>2019-08-27 19:50:20</div>
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	<b>小棒叮</b> 作为一个中学生，之前在少儿编程编程的网站对Python还是有一定的了解的，毫不犹豫地买了这本Python入门指南，就是想对Python有更深入的了解，奋斗吧，少年！ <div>👍 6    回复</div> <div>2019-08-22</div>
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	

← 慕课专栏	☰ 你的第一本Python基础入门书 / 01 开篇词：你为什么要学 Python？	
	回复	2019-08-23 22:04:38
目录		
第 1 章 入门准备		
01 开篇词：你为什么要学 Python？ <a href="#">最近阅读</a>		
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		