极客学院
jikexueyuan.com

互联网+职业技能系列

职业入门 | 基础知识 | 系统进阶 | 专项提高

O2O 学习新模式

Python 基础实例教程

微课版

Python Foundation

韦玮 著

- 73 个在线**微课视频**配合图书**同步讲解**
- **精心挑选**简单实用的案例，即**学**即**练**
- 附**赠**案例**源代码**

扫码关注图书
获取配套资源

中国工信出版集团

人民邮电出版社
POSTS & TELECOM PRESS

极客学院
jikexueyuan.com

互 联 网 + 职 业 技 能 系 列
职业入门 | 基础知识 | 系统进阶 | 专项提高

Python

基础实例教程

微课版

Python Development

韦玮 著

人 民 邮 电 出 版 社
北 京

图书在版编目（CIP）数据

Python基础实例教程 : 微课版 / 韦玮著. -- 北京 : 人民邮电出版社, 2018.9
（互联网+职业技能系列）
ISBN 978-7-115-48713-1

Ⅰ. ①P… Ⅱ. ①韦… Ⅲ. ①软件工具－程序设计－教材 Ⅳ. ①TP311.56

中国版本图书馆CIP数据核字(2018)第137030号

内 容 提 要

本书较为全面地介绍了 Python 编程相关的知识。全书共 14 章，包括 Python 开发环境搭建与入门、语法基础、运算符与表达式、控制流、函数、模块、数据结构、常见算法实例、面向对象程序设计、异常处理、文件操作、标准库与其他应用、远程操控计算机项目、腾讯动漫爬虫项目等内容。

本书可以作为高校计算机类专业的教材，也适合作为编程开发人员、计算机销售技术支持的专业人员和广大计算机爱好者的自学参考书。

◆ 著　　韦　玮
责任编辑　左仲海
责任印制　马振武
◆ 人民邮电出版社出版发行　　北京市丰台区成寿寺路 11 号
邮编　100164　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
固安县铭成印刷有限公司印刷
◆ 开本：787×1092　1/16
印张：15.75　　2018 年 9 月第 1 版
字数：421 千字　　2018 年 9 月河北第 1 次印刷

定价：49.80 元

读者服务热线：(010)81055256　印装质量热线：(010)81055316
反盗版热线：(010)81055315
广告经营许可证：京东工商广登字 20170147 号

前言
Foreword

关于本书

Python 是一门非常适合新手入门的编程语言，语法简洁，易于上手，并且功能十分强大，在许多领域都有着非常广泛的应用。

编者编写本书是希望能够帮助一些没有编程基础或者编程基础薄弱的读者建立起程序设计相关的知识体系。为此，编者为本书配上了全套的视频教程，希望可以让读者学习起来更加轻松，更容易地理解本书中的内容。同时，本书的编排遵循由简到难的原则，知识的难度逐步加大，层层递进。本书的编程实例非常丰富，读者可以跟着书中的实例进行学习。对于初学者来说，在实践中学习不会感觉过于枯燥。

尽管本书已经尽力做到通俗易懂、易于学习，但学习仍然是一件艰苦的事情，希望广大的读者朋友可以坚持下去，早日学会 Python 编程。

勘误与支持

由于编者水平有限，书中难免有一些不足或疏漏之处，恳请各位读者不吝指正。读者可以通过微博“@韦玮 pig”或微信公众平台“正版韦玮”（可以直接扫描最下方二维码添加）反馈相关建议，也可以直接向邮箱 ceo@iqianyue.com 发送邮件（标题注明：勘误反馈-书名）。期待能够收到读者的意见和建议。

致谢

感谢这么久以来一直支持我的学生们！平常公司的工作比较忙，如果没有你们一直以来的支持，在业余时间完成课程的录制及书籍的写作确实太难。你们的支持与包容是我在这个领域一直走下去的动力，非常感谢大家！

特别感谢我的女友！因为编写这本书，少了很多陪你的时间，感谢你的不离不弃与理解，同时，也感谢你帮我完成书稿的校对工作，谢谢你的付出与支持。

特别感谢远方的父母、叔叔、姐姐、爷爷，也特别感谢所有支持我的朋友们。

配套资源下载

读者可以通过以下微信公众平台，或者登录人民邮电出版社教育社区（www.ryjiaoyu.com）下载相关资源。

编者

2018 年 3 月

目录
Contents

CHAPTER 01

第1章

初识Python

■ Python 是一门面向对象的、解释型的编程语言，具有语法简洁、易读、功能强大等特点，对于初学者来说，非常容易上手。而且，使用 Python 可以完成非常多的需求，比如开发网络爬虫，进行自动化运维、自动化测试、逆向编程、量化分析等。所以，Python 这门语言获得了大量 IT 从业人员以及编程爱好者的喜爱。本章会对 Python 的基础进行介绍，让读者对 Python 有一个快速的认识。

十分钟快速了解 Python

1.1 快速了解 Python

Python 是一门非常优秀的语言，于 1989 年由吉多·范罗苏姆（Guido van Rossum）创立，拥有语法简洁、易于学习、功能强大等多种优势与特点，所以非常受程序员的欢迎。目前 Python 在最流行的 10 种编程语言排行榜中排名第 1。

1.1.1 Python 的起源与背景

1989 年圣诞节期间，荷兰人吉多·范罗苏姆打算开发一门新的语言作为 ABC 语言的继承，随后便以自己非常喜欢的一个喜剧团 Monty Python 的名字中的 Python 为这门新语言命名。就这样，在吉多·范罗苏姆的努力下，Python 语言诞生了。从一开始，吉多·范罗苏姆在 Python 语言的设计中就特别在意易读性与可扩展性。比如强制缩进这方面的设计就参照了当时的 ABC 语言，非常有利于提升程序的易读性。

当然，强制缩进这方面的内容目前还存在着比较大的争议，有些人认为强制缩进让编写变得更加麻烦，而有些人认为强制缩进可以让代码更加整洁。笔者觉得强制缩进是利大于弊的。虽然在写程序的时候，如果不注意程序的缩进，会导致程序出现错误，稍微麻烦了一点，但是缩进可以让程序更加美观，在阅读程序的时候可以一目了然。而且，缩进也是有规律的，最大的规律就是同一层级的代码在同一个缩进幅度上。这样做可以让程序的层次结构变得非常分明，尤其在代码多的时候，如果层次分明，可以更方便开发与管理程序。同样，读者在写 Python 程序的时候，只需要把握住这一规律，就会发现强制缩进非但不麻烦，而且还会提升开发效率。

在可扩展性方面，Python 语言做得也是相当不错的。比如可以将一些常用的功能写成.py 文件，然后放到 Python 安装目录的 Lib 目录下，这样该文件就成了一个 Python 的模块，此时若想用对应的功能，直接导入对应的模块即可。比如对应的.py 文件的文件名为 a.py，在将该文件封装为模块之后，直接通过 import a 就可以导入该文件，然后就可以使用该文件中所实现的功能了。除此之外，也可以使用 C 语言去写一些程序文件，写好了之后只需要将对应的 C 语言文件编译为.so 文件，随后就可以直接在 Python 中引入对应的.so 文件并使用了。正因为 Python 语言在可扩展性方面做得非常好，所以使用 Python 来实现各种功能都非常适合，这让 Python 具备了强大的功能。

在 1991 年的时候，Python 第一版正式发行。第一版 Python 就具备了核心数据类型、函数、模块、异常处理及面向对象等方面的内容。

在接下来的发展中，Python 语言获得了很多 Python 用户的支持与改进。开始的时候，Python 用户以邮件列表（maillist）的方式进行沟通和开发，不同的用户使用 Python 开发出一些功能或改进之后，会将这些改进及新的功能发送给吉多·范罗苏姆。如果吉多·范罗苏姆觉得这些改进或新的功能非常有用，则会将这些改进或新的功能添加到 Python 或者 Python 的标准库中。之前提到过，Python 的可扩展性非常好，所以，当用户的改进或新的功能添加到 Python 之后，Python 可以继续保留原有的功能，也可以很轻松地对接新的功能。

随后，Python 用户越来越多，Python 社区也越来越大，Python 社区后来也拥有了自己的网站（python.org），之后 Python 的开发与改进方式也由原先通过邮件列表的方式逐渐向开源的方式转变。

目前，Python 已经拥有了大量的模块，通过不同的模块，可以实现各式各样的功能。读者可以通过 Python 社区的 PyPI 查找别人开发的模块并使用。当然，用户开发或改进的某个新功能，也可以通过 PyPI 提交上去，供别的开发者使用。

Python 发展到今天，已经深受广大程序开发者的喜爱，应用在各个行业上，比如常见的爬虫、数据挖掘、人工智能等领域，Python 都有着极其广泛的应用。

如图 1-1 所示，2017 年，Python 在最流行的 10 种编程语言排行榜中排名第 1，也在逐渐地影响

着更多的程序员。

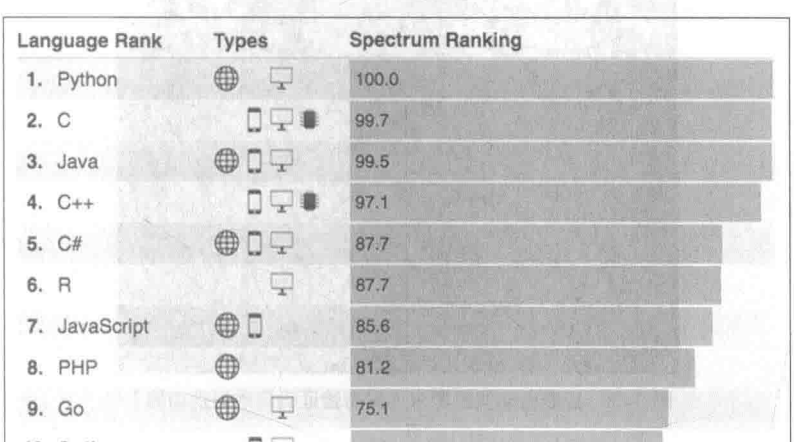

图 1-1　2017 年最流行的 10 种编程语言

1.1.2　Python 的功能

由于 Python 的可扩展性非常好，所以 Python 可以实现的功能也非常多。

就目前来说，经常使用 Python 来处理的领域有简单脚本编程、Web 系统开发、爬虫数据采集、数据分析与挖掘、自动化运维等。

除此之外，Python 还可以实现很多其他的功能。例如游戏开发、黑客逆向编程、网络编程等方面，使用 Python 来实现也是非常适合的。

在使用 Python 开发程序时，尤其是在实现某个专业方向的功能时，通常会使用到 Python 的模块。一般来说，Python 的自带模块（即标准库）就已经非常丰富了，但如果某个自带模块无法实现更深层次的功能，开发者还可以选择使用第三方模块进行开发。如果还无法满足需求，也可以自己开发一些程序封装成模块使用，这些都是很容易实现的。也正因为如此，使得 Python 能够适应多个专业领域的开发。当然，在初学 Python 的时候，并不需要大家对 Python 所有的标准库都非常熟悉，只需要关注与自己专业方向相关的标准库即可。例如，如果主要做网络爬虫，可以重点关注 urllib（自带模块）、re（自带模块）、Scrapy（第三方模块），对于其他模块，可以有选择地掌握。总之，一切以需求为导向。

为了让大家能够更好地理解 Python 的功能，笔者将为大家展示一些自己用 Python 开发实现的项目案例。

图 1-2 所示是使用 Python 实现的乘法口诀表，可以使用循环自动地输出。

```
1*0=0
2*0=0  2*1=2
3*0=0  3*1=3  3*2=6
4*0=0  4*1=4  4*2=8  4*3=12
5*0=0  5*1=5  5*2=10  5*3=15  5*4=20
6*0=0  6*1=6  6*2=12  6*3=18  6*4=24  6*5=30
7*0=0  7*1=7  7*2=14  7*3=21  7*4=28  7*5=35  7*6=42
8*0=0  8*1=8  8*2=16  8*3=24  8*4=32  8*5=40  8*6=48  8*7=56
```

图 1-2　使用 Python 实现的乘法口诀表

图 1-3 所示是使用 Python 实现的豆瓣自动登录网络爬虫，会自动登录豆瓣网（假如遇到验证码，会自动识别验证码再自动登录），然后爬取个人中心页面日志数据。

```
D:\loginpjt>scrapy crawl loginspd --nolog
此时没有验证码
登录中…
此时已经登录完成并爬取了个人中心的数据
网页标题是：
韦老师

第1篇文章的信息如下：
文章标题为：测试2
文章发表时间为：2016-10-03 19:31:40
文章内容为：这是测试2的正文内容。
文章链接为：https://www.douban.com/note/584776354/
----------
第2篇文章的信息如下：
文章标题为：test7799
文章发表时间为：2016-10-03 11:46:57
文章内容为：mytest this is my test 779988
文章链接为：https://www.douban.com/note/584726594/
----------
```

图 1-3　豆瓣自动登录爬虫（具有验证码自动识别功能）

图 1-4 所示是使用 Python 实现的智能预测课程销量的一个程序，使用的是人工神经网络算法。当然，由于实验数据及时间有限，当前的 loss 值为 0.3818，相对比较高，所以准确率不算太高，即此时有 61.82%的概率是预测准确的，此时加多训练次数以及加大训练数据数量即可改善准确率。从图 1-4 中也可以看到，随着训练次数的增加，loss 逐渐减小，即准确率逐渐提高。

```
Epoch 997/1000
29/29 [==============================] - 0s - loss: 0.3822
Epoch 998/1000
29/29 [==============================] - 0s - loss: 0.3821
Epoch 999/1000
29/29 [==============================] - 0s - loss: 0.3820
Epoch 1000/1000
29/29 [==============================] - 0s - loss: 0.3818
29/29 [==============================] - 0s
0.8620689655172413
3/3 [==============================] - 0s
第1门课程的销量预测结果为：高
第2门课程的销量预测结果为：高
第3门课程的销量预测结果为：低
```

图 1-4　使用 Python 实现的预测课程销量的程序执行结果

除此之外，还能使用 Python 做很多有趣的事情。

当然，要想使用 Python 实现各种各样的功能，首先需要把 Python 基础打好，本书会为大家全面地呈现 Python 的基础知识。基础打好之后，方可继续研究深层次的内容。

Python 简介及特色

1.1.3　Python 的优势与特色

Python 作为一门流行的编程语言，有着非常多的优势与特色。下面就来介绍一下 Python 基本的优势与特色。

1. Python 的优势

Python 语言有着三大显著的优势，即：

（1）简单易学。

（2）功能强大。

（3）支持面向对象。

首先，如果想入门 Python，相对来说是非常容易的，因为 Python 的编程风格非常简洁。举个例子，想定义一个变量并赋值为 19，然后输出该变量的值，如果使用 C++进行编写，需要通过如下程序

实现：

```
#include <iostream>
using namespace std;

int main() {
    int i=19;
    cout<<i;
    return 0;
}
```

而如果使用 Python 实现，以下程序即可：

```
i=19
print(i)
```

都是输出 19。

从上面的例子可以看出，相对来说，Python 的编程风格是非常简洁的，正因为如此，读者很快就能学会 Python。

Python 的语法如此简洁，是否意味着 Python 的功能不够强大呢？

答案当然是否定的。Python 的功能非常强大，几乎是一门全能的语言，总结来说，Python 可以应用在以下方面。

（1）系统编程。

（2）GUI 编程。

（3）开发网络爬虫。

（4）Web 开发。

（5）数据分析与挖掘。

（6）机器学习领域。

（7）游戏开发。

（8）自动化运维。

Python 的第三个优势就是支持面向对象编程，这个优势可以让 Python 在开发大型项目的时候变得非常方便。

2. Python 的特色

Python 常见的一些特点如下。

（1）大小写严格区分。

（2）简单、易学、支持面向对象。

（3）开源。

（4）库非常丰富。

（5）跨平台使用。

（6）解释性语言。

（7）高级语言。

这些特点大家在后续的学习过程中会逐渐感受到，前两个特点不用过多阐述，接下来重点介绍一下后 5 个特点。

Python 是开源的，所谓“开源”，简单理解就是开放源代码。正因为其是开源的，所以可以让更多的人传播和使用 Python，并且能够更好地发现其中的 Bug 并修复，这大大促进了 Python 的发展。

Python 的库是非常丰富的，所谓“库”，读者可以理解为是一系列 Python 功能的封装，比如要使用 Python 开发一个网络爬虫，那么就可以使用网络爬虫相关的库。首先导入 urllib 库，然后直接进行网络爬虫的编写，因在 urllib 库中封装了大量与网络爬虫相关的功能。再比如，需要使用 Python 实现一些与操作系统相关的功能，如运行某个 shell 命令，此时可以使用 Python 的 os 库进行实现，因为在 os 库中封装了大量与操作系统相关的功能。

接触 Python 后就会发现，正因为 Python 的库非常丰富，所以使用 Python 来实现各种各样的功能就变得非常方便了。

另外，跨平台使用也是 Python 的一大特点。比如，在 Windows 操作系统写好的 Python 程序，可以不加修改或者只进行少量修改，就能够在 Linux 系统、Mac 系统及其他操作系统中运行。这一点对于程序开发来说是极为方便的。

此外，Python 是一门解释性语言，同时也是一门高级语言，这是它的另外两大特点。

解释性语言区别于编译型语言，解释性语言是在程序运行的时候将程序翻译成机器语言，而编译型语言则需要在程序执行之前进行一个编译的过程，统一地将程序转换为机器语言，然后执行。按常规来说，解释性语言的执行速度一般会比编译型语言慢，但是，学到后面会发现，Python 程序在执行的时候会生成一个跟程序对应的 PYC 格式的文件，这样可以大大提高程序运行的速度。关于 PYC 格式的文件，此处不需要深入理解，因为在后面会详细介绍到。

高级语言与低级语言不同，一般来说，与人类思维方式更接近的语言称为高级语言，而与机器的运行方式（二进制）更接近的语言称为低级语言。两种类型的语言各有优势，比如，使用高级语言编程，编写的速度自然会更快一些，而使用低级语言实现相同的功能，则相对慢一些。但是，在程序运行的时候，由于低级语言更接近于机器的习惯，而高级语言需要经过解释或编译的过程转换为机器语言，之后再交由机器执行，所以，高级语言的运行速度一般来说会比低级语言的运行速度慢。

这里为大家介绍了 Python 的三大优势与七大特点，当然，这些并不代表 Python 的所有优势与特点，其他的优势及特点大家可以在后续的学习中逐步领会。

Windows 下安装 Python

1.2 在 Windows 下搭建 Python 开发环境

要想学习 Python 程序开发，首先需要搭建好 Python 的开发环境。所以，需要在自己的计算机上安装好 Python，并配置好相应的开发环境。

在此，先为大家介绍如何在 Windows 操作系统中安装 Python 并配置相关的开发环境。

1.2.1 操作系统的选择

考虑到目前大部分读者使用的都是 Windows 操作系统的计算机，而 Python 的跨平台能力又非常强，所以对于学习 Python 的读者来说，操作系统并不是学习 Python 的关键影响因素，所以本书会选择大部分读者熟悉的 Windows 操作系统进行编写。

在实际的工作环境中，编程人员常常在 Windows 以及 Mac 计算机中进行 Python 程序的开发，当然也会有一部分公司喜欢用 Linux 系统进行 Python 程序的开发，所以大家在掌握了 Python 程序开发的知识之后，未来有机会也可以学习一下 Linux 系统的使用。

一般来说，在 Python 自动化运维领域，应该优先选用 Linux 操作系统进行 Python 程序的开发。而在其他 Python 应用的领域，比如网络爬虫、Web 开发等方面，使用哪种系统进行 Python 程序开发并不是那么重要，这也是本书为什么选择 Windows 操作系统进行编写的原因之一。

1.2.2 在 Windows 下安装 Python

如果希望在 Windows 计算机上搭建 Python 开发环境，总体上可以按照以下步骤进行。

（1）选择 Python 版本。

（2）下载对应版本的 Python。

（3）在计算机上安装好 Python。

（4）配置好环境变量。

（5）选择一款合适的编辑器（可选）。

下面分别对这些步骤进行详细的讲解。

1. 选择 Python 的版本

目前，Python 版本主要分为两种，一种是 Python 2.X，另外一种是 Python 3.X。这两种版本的兼容性并不是太好，所以还是推荐大家使用 Python 3.X 进行学习，毕竟 Python 3.X 在各相关公司用得越来越多，也必将是未来的发展趋势。

至于使用 Python 3.X 系列下面哪个具体的版本，并不是那么重要。一般来说，Python 3.4 及以上的版本，差别并不是太大，本书选择的版本是 Python 3.5.2，当然，读者也可以安装 Python 3.6 及以上的版本进行学习，影响不大。

2. 下载对应版本的 Python

读者可以打开 Python 官网，然后选择 Downloads 下面的 Windows 选项，如图 1-5 所示。

图 1-5　在官网选择 Windows 版本的 Python 进行下载

然后，在出现的页面中可以选择相关的版本，比如选择 Python 3.5.2 版本，如图 1-6 所示。

图 1-6　选择具体的 Python 版本

这里有很多个文件链接，此时只需要重点关注以 executable installer 结束的文件即可，代表对应的文件为可执行的安装文件。

在图 1-6 中，出现了两个相关的文件，即：

（1）Windows x86 executable installer。

（2）Windows x86-64 executable installer。

这两个文件中，（1）为 32 位的安装包（即 x86 字样的），（2）为 64 位的安装包（即 x86-64 字样的），所以此时需要查看读者的计算机的位数，如果计算机为 64 位，选择文件（2）进行下载即可。

3. 安装下载的 Python

双击下载好的文件（如果无法打开或出现问题，可以右键单击图标，选择“以管理员身份运行”命令，即可解决问题），会出现图 1-7 所示的界面。

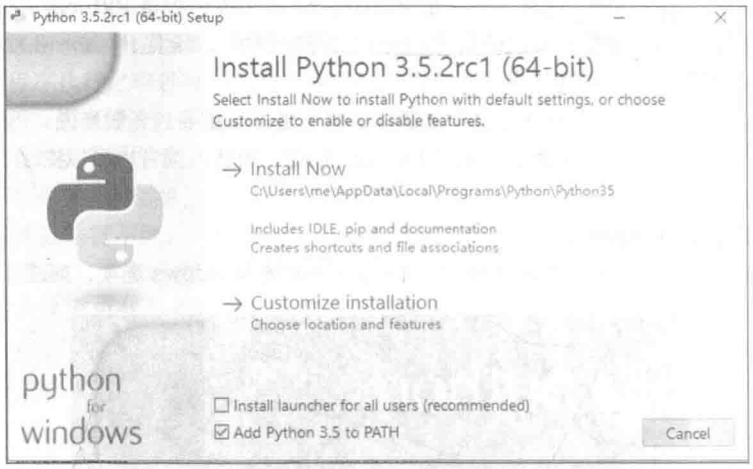

图 1-7　安装 Python 的界面

在该界面中，可以将下方的“Add Python 3.5 to PATH”复选框勾选上，此时会对相关的环境变量进行自动配置。此外，如果不想为所有用户安装 Python，也可以取消选择“Install launcher for all users（recommended）”复选框，如图 1-7 所示。

随后，可以单击图 1-7 中的“Customize installation”，会出现图 1-8 所示的界面。

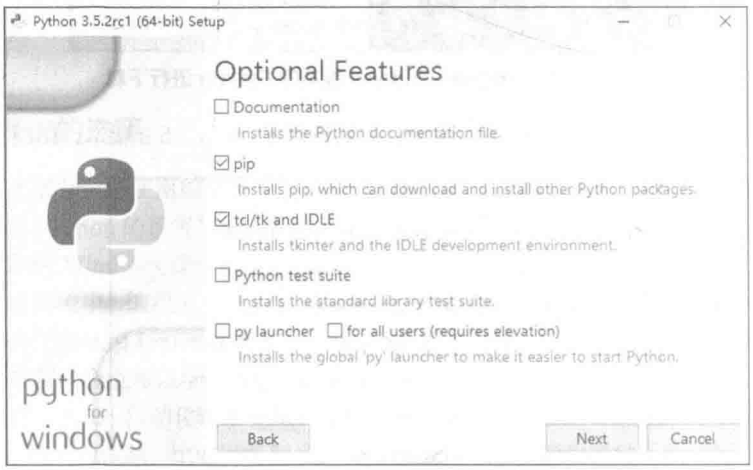

图 1-8　选项配置界面

此时可以勾选“pip”与“tcl/ tk and IDLE”复选框。pip 工具可以极大地方便模块的安装；IDLE 则为默认的 Python 编辑器；其他的选项部分，如果不需要可以不必勾选。这样可以节省安装时间，随后单击“Next”按钮，会出现图 1-9 所示的界面。

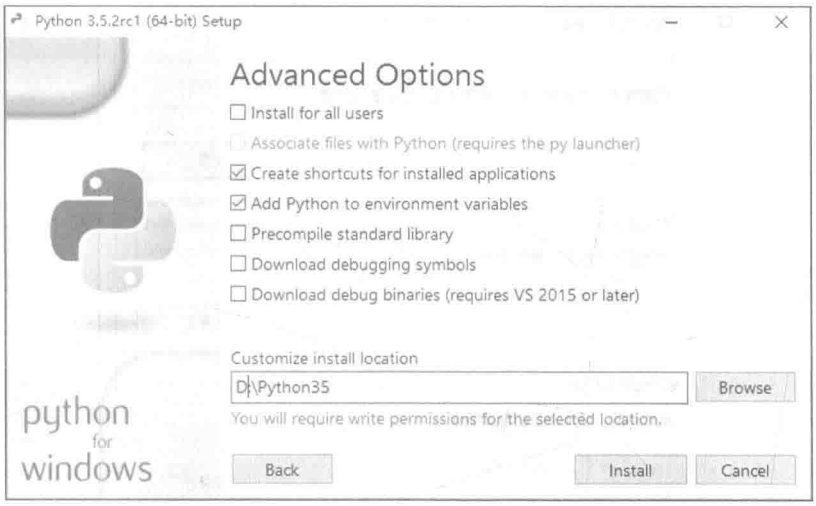

图 1-9　设置 Python 的安装目录

在图 1-9 所示的界面中，可以设置 Python 的安装目录，例如可以将路径设置在 D 盘下的 Python35 文件夹中。选项设置方面，可以将第 3、4 个复选框勾选上，取消其他复选框的勾选，如图 1-9 所示，然后，单击“Install”按钮即可安装成功。

4．环境变量的配置

由于在安装的时候勾选了自动添加环境变量，如果不出意外，环境变量此时已经添加好了。但有时由于系统或者其他原因，环境变量可能自动添加不上，所以最好检查一下。

首先，在计算机左下角的运行框内输入“环境变量”后进行搜索，然后在出现的匹配结果中选择“编辑系统环境变量”，如图 1-10 所示。

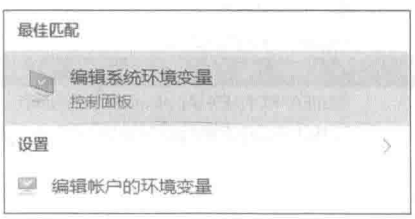

图 1-10　选择“编辑系统环境变量”

此时会出现图 1-11 所示的对话框。

此时，需要单击图 1-11 中的“环境变量”按钮，即可出现环境变量的相关配置对话框，如图 1-12 所示。

然后选择“me 的用户变量”列表框中的“PATH”变量，只需要单击即可选中，如图 1-12 所示。

在此编辑 PATH 环境变量的目的，是为了告诉系统 Python 安装在了什么地方，否则系统无法知道 Python 在什么地方，所以如果环境变量没有配置好，在 cmd 命令行下输入“python”，会出现相关错误。

选中“PATH”变量之后，单击图 1-12 中的“编辑”按钮，则会出现图 1-13 所示的对话框。