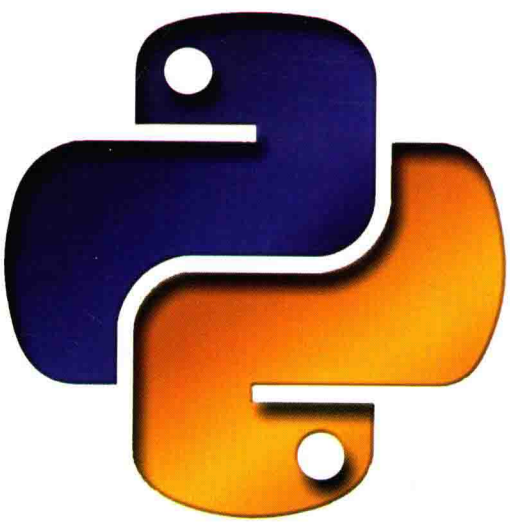
高等教育规划教材

免费提供
教学 PPT、源代码、教学大纲、
教案、习题答案等全套教学资源

Python 程序设计
基础与应用

董付国 著



机械工业出版社
CHINA MACHINE PRESS

高等教育规划教材

Python 程序设计基础与应用

董付国　著

机械工业出版社

本书是一本系统介绍 Python 程序开发与应用的教程。本书共 15 章，主要包括 Python 编程基础（1～11 章）和 Python 开发应用（12～15 章）两部分内容，编程基础部分通过众多案例对 Python 程序设计的概念加以解释，开发应用部分介绍了 tkinter 编程、网络爬虫、数据分析和数据可视化 4 个方面的 Python 核心应用。本书全部代码适用于 Python 3.5、Python 3.6 以及更高版本。

本书可以作为非计算机专业研究生、本科、专科程序设计课程教材，也可作为计算机专业本、专科程序设计基础课程教材，以及 Python 爱好者自学用书。

本书配有教学资源（包括 PPT、源码、大纲、教案、习题答案），需要的教师可登录 www.cmpedu.com 免费注册，审核通过后下载，或联系编辑索取（QQ：2966938356，电话：010-88379739）。

图书在版编目（CIP）数据

Python 程序设计基础与应用 / 董付国著. —北京：机械工业出版社，2018.9
高等教育规划教材
ISBN 978-7-111-60617-8

Ⅰ. ①P… Ⅱ. ①董… Ⅲ. ①软件工具－程序设计－高等学校－教材
Ⅳ. ①TP311.561

中国版本图书馆 CIP 数据核字（2018）第 175990 号

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）
策划编辑：王 斌 责任编辑：王 斌
责任校对：张艳霞 责任印制：常天培

北京富博印刷有限公司印刷厂印刷

2018 年 9 月第 1 版 • 第 1 次印刷
184mm×260mm • 14.75 印张 • 10 插页 • 362 千字
0001—4000册
标准书号：ISBN 978-7-111-60617-8
定价：49.00元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务
服务咨询热线：（010）88379833
读者购书热线：（010）88379649

网络服务
机 工 官 网：www.cmpbook.com
机 工 官 博：weibo.com/cmp1952
教育服务网：www.cmpedu.com
金 书 网：www.golden-book.com

封面无防伪标均为盗版

前　言

Python 语言由 Guido van Rossum 于 1991 年推出了第一个公开发行版本，之后迅速得到了各行业人士的青睐。经过 20 多年的发展，Python 语言已经渗透到统计分析、移动终端开发、科学计算可视化、系统安全、逆向工程、软件测试与软件分析、图形图像处理、人工智能、机器学习、深度学习等几乎所有专业和领域，在黑客领域更是多年来一直拥有霸主地位。与此同时，Python 语言在各大编程语言排行榜上的位次也是逐年上升，被 TIOBE 网站评为 2010 年年度语言；在IEEE Spectrum 2017 编程语言排行榜上则名列榜首，排在了第一位。

Python 是一门免费、开源、跨平台的高级动态编程语言，支持命令式编程、函数式编程，完全支持面向对象程序设计，拥有大量功能强大的内置对象、标准库，以及涉及各行业领域的扩展库，使得各领域的工程师、科研人员、策划人员和管理人员能够快速实现和验证自己的思路、创意或者推测，还有更多人喜欢用 Python 写个小脚本来完成自己工作中的一些小任务。在有些编程语言中需要编写大量代码才能实现的功能，在 Python 中只需要几行代码，大幅度减少了代码量，更加容易维护。Python 用户只需要把主要精力放在业务逻辑的设计与实现上，在开发速度和运行效率之间达到了完美的平衡，其精妙之处令人击节赞叹。

一个好的 Python 程序不仅是正确的，更是简洁的、直观的、漂亮的、优雅的、方便人们阅读的，整个代码处处体现着美，让人赏心悦目。Python 代码对布局要求非常严格，尤其是使用缩进来体现代码的逻辑关系，这一点硬性要求非常有利于学习者和程序员养成一个良好的、严谨的习惯。除了能够快速解决问题之外，代码布局要求严格也是 Python 被广泛选作教学语言的重要原因。

早在多年前 Python 就已经成为卡耐基梅隆大学、麻省理工学院、加州大学伯克利分校、哈佛大学、多伦多大学等国外很多大学计算机专业或非计算机专业的程序设计入门教学语言。近几年来国内有几百所高等院校的多个专业陆续开设了 Python 程序设计有关课程，并且这个数量还在持续快速增加。目前来看，选择使用 Python 作为程序设计入门教学语言或者作为各专业扩展课程，无疑是一个非常明智的选择。

内容组织与阅读建议

本书共 15 章，主要包括 Python 编程基础（1～11 章）和 Python 开发应用（12～15 章）两部分内容，全部代码适用于 Python 3.5、Python 3.6 以及更高版本。

第 1 章　概述。简单介绍 Python 语言与版本、编程规范，扩展库安装方法，以及标准库对象与扩展库对象的导入与使用。

第 2 章　内置对象运算符、表达式、关键字。讲解 Python 常用内置对象、运算符与表达式、常用内置函数和 Python 关键字。

第 3 章　Python 序列结构。讲解列表、列表推导式、切片操作，元组与生成器表达式，字典，集合和序列解包。

第 4 章　选择结构与循环结构。讲解条件表达式的常见形式，单分支、双分支、多分支选择结构以及嵌套的选择结构，for 循环与 while 循环，break 与 continue 语句。

第 5 章　函数。讲解函数定义与调用语法，不同类型的函数参数，参数传递的序列解包，

变量作用域，lambda 表达式，生成器函数。

第 6 章　面向对象程序设计。讲解类的定义与使用，数据成员与成员方法、属性，继承，特殊方法与运算符重载。

第 7 章　字符串。讲解字符串编码格式，转义字符与原始字符串，字符串格式化的不同形式，字符串常用方法与操作，字符串常量，以及扩展库 jieba 和 pypinyin 的用法等。

第 8 章　正则表达式。讲解正则表达式语法，正则表达式模块 re 的用法和 match 对象等。

第 9 章　文件内容操作。讲解文件操作基本知识，文本文件内容操作方法，常用的二进制文件读写模块，以及 Excel、Word 等常见类型文件的操作。

第 10 章　文件与文件夹操作。讲解 os、os.path 与 shutil 这 3 个模块的用法，以及递归遍历并处理文件夹的原理。

第 11 章　异常处理结构。介绍异常的常见表现形式，常用异常处理结构，以及断言语句与上下文管理语句。

第 12 章　tkinter 编程案例。通过模拟用户登录、选择类组件应用、简单画图程序、电子时钟、屏幕颜色选择器、抽奖式提问程序、简易计算器程序、定时自动关闭的窗口等案例演示 Python 标准库 tkinter 的用法。

第 13 章　网络爬虫入门与应用。介绍 HTML 和 JavaScript 基础，标准库 urllib 以及扩展库 scrapy、BeautifulSoup4、requests 和 selenium 在网络爬虫程序设计中的应用。

第 14 章　Python 数据分析与处理。讲解使用 pandas 库进行数据分析的基本操作，数据分析案例与 pandas 的应用。

第 15 章　数据可视化。介绍使用 Python 扩展库 matplotlib 进行数据可视化相关的技术，包括折线图、散点图、饼状图、柱状图和雷达图的绘制，以及坐标轴、图例等设置。

本书适用读者

本书可以作为（但不限于）：非计算机专业研究生、本科、专科程序设计课程教材；计算机专业程序设计基础课程教材；Python 爱好者自学用书。

配套资源

本书为选用教材的老师提供教学 PPT、源码、大纲、教案、习题答案等全套教学资源，可通过微信公众号“Python 小屋”获取，或发送邮件至 dongfuguo2005@126.com 与作者联系获取；也可通过机械工业出版社相应渠道获取（见版权页内容简介）。

致谢

首先感谢父母的养育之恩，在当年那么艰苦的条件下还坚决支持我读书，没有让我像其他同龄的孩子一样辍学。感谢姐姐、姐夫多年来对我的爱护以及在老家对父母的照顾，感谢善良的弟弟、弟媳在老家对父母的照顾。当然，最应该感谢的是妻子和孩子对我这个工作狂人的理解和体谅。

感谢每一位读者，感谢您在茫茫书海中选择了本书，衷心祝愿您能够从本书中受益，学到真正需要的知识。同时也期待每一位读者的热心反馈，随时欢迎您指出书中的不足，并通过微信公众号“Python 小屋”或电子邮箱 dongfuguo2005@126.com 与作者沟通和交流。

董付国　于山东烟台

2018 年 3 月

目　录

第1章 Python概述

Python 语言以快速解决问题而著称，其特点在于提供了丰富的内置对象、运算符和标准库对象，而庞大的扩展库更是极大增强了 Python 的功能，大幅度拓展了 Python 的用武之地，其应用几乎已经渗透到了所有领域和学科。本章将介绍 Python 语言的特点、版本、编码规范、扩展库的安装、标准库对象与扩展库对象的导入和使用。

本章学习目标

- 了解 Python 语言版本
- 熟悉 Python 开发环境
- 了解 Python 编码规范
- 掌握扩展库安装方式
- 掌握标准库对象与扩展库对象的导入和使用

1.1 Python 语言简介

Python 语言的名字来自于一个著名的电视剧“Monty Python's Flying Circus”，Python 之父 Guido van Rossum 是这部电视剧的狂热爱好者，所以把他设计的语言命名为 Python。

Python 是一门跨平台、开源、免费的解释型高级动态编程语言，是一种通用编程语言。除了可以解释执行之外，Python 还支持将源代码伪编译为字节码来优化程序提高加载速度并对源代码进行一定程度的保密，也支持使用 py2exe、pyinstaller、cx_Freeze 或其他类似工具将 Python 程序及其所有依赖库打包成为各种平台上的可执行文件；Python 支持命令式编程和函数式编程两种方式，完全支持面向对象程序设计，语法简洁清晰，功能强大且易学易用，最重要的是拥有大量的几乎支持所有领域应用开发的成熟扩展库。

Python 语言拥有强大的“胶水”功能，可以把多种不同语言编写的程序融合到一起实现无缝拼接，更好地发挥不同语言和工具的优势，满足不同应用领域的需求。Python 诞生以来，不到 30 年的时间里，已经渗透到统计分析、移动终端开发、科学计算可视化、系统安全、逆向工程与软件分析、图形图像处理、人工智能、机器学习、游戏设计与策划、网站开发、数据爬取与大数据处理、密码学、系统运维、音乐编程、影视特效制作、计算机辅助教育、医药辅助设计、天文信息处理、化学与生物信息处理、神经科学与心理学、自然语言处理、电子电路设计、电子取证、树莓派（Raspberry Pi，为学习计算机编程教育而设计，只有信用卡大小的微型电脑）开发等几乎所有专业和领域，在黑客领域更是多年来一直拥有霸主地位。

1.2 Python 版本简介

Python 官方网站同时发行和维护着 Python 2.x 和 Python 3.x 两个不同系列的版本，并且版本更新速度非常快。目前常用版本分别是 Python 2.7.14、Python 3.4.8、Python 3.5.5 和 Python 3.6.4。另外，本书定稿时 Python 3.7.0 已经推出 beta2 测试版本，预计在本书出版之前会发布正式版本。Python 2.x 和 Python 3.x 这两个系列的版本之间很多用法是不兼容的，除了基本输入/输出方式有所不同，很多内置函数和标准库对象的用法也有非常大的区别，适用于 Python 2.x 和 Python 3.x 的扩展库之间更是差别巨大，这也是旧系统进行版本迁移时最大的障碍。

Python 3.x 的设计理念更加合理、高效和人性化，代码开发和运行效率更高，2015 年年底开始 Python 3.x 就已经呈现出全面普及和应用的趋势，越来越多的扩展库也以非常快的速度推出了与最新 Python 版本相适应的版本。另外，Python 官方早在 2016 年就已经宣布，最迟到 2020 年将会全面放弃 Python 2.x 的维护和更新。所以，如果你正在使用 2.x 系列，那么最好尽快转换成 3.x 并且选择较高的版本。如果刚刚开始接触 Python，那么一定要毫不犹豫地选择最新的 3.x 版本。

1.3 Python 开发环境安装与配置

除了 Python 官方安装包自带的 IDLE，还有 Anaconda3、PyCharm、Eclipse、zwPython 等大量开发环境。相对来说，IDLE 稍微简陋一些，但也提供了语法高亮（使用不同的颜色显示不同的语法元素，例如，使用绿色显示字符串、橙色显示 Python 关键字、紫色显示内置函数）、交互式运行、程序编写和运行以及简单的程序调试功能。其他 Python 开发环境则是对 Python 解释器主程序进行了不同的封装和集成，使得代码的编写和项目管理更加方便一些。本节对 IDLE 和 Anaconda3 这两个开发环境进行简单介绍，但书中所有代码也同样可以在 PyCharm 等其他开发环境中运行。

按照惯例，本书中所有在交互模式运行和演示的代码都以 IDLE 交互环境的提示符“>>>”开头，在运行这样的代码时，并不需要输入提示符“>>>”。而书中所有不带提示符“>>>”的代码都表示需要写入一个程序文件并保存和运行。

1.3.1 IDLE

IDLE 应该算是最原始的 Python 开发环境之一，没有集成任何扩展库，也不具备强大的项目管理功能。但也正是因为这一点，使得开发过程中的一切尽在自己掌握中，深得资深 Python 爱好者喜爱，成为 Python 内功修炼的重要途径。

在 Python 官方网站 https://www.python.org/下载最新的 Python 3.6.x 安装包或 Python 3.7.x（根据自己计算机操作系统选择 32 位或 64 位）并安装（建议安装路径为 C:\Python36 或 C:\Python37）后，在开始菜单中可以打开 IDLE，如图 1-1 所示，然后看

到的界面就是交互式开发环境，如图 1-2 所示。

图 1-1 “开始”菜单

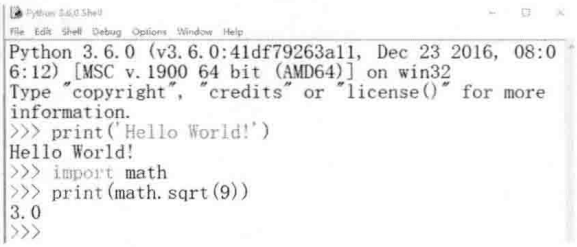


图 1-2 IDLE 交互式开发界面

在交互式开发环境中，每次只能执行一条语句，当提示符“>>>”再次出现时方可输入下一条语句。普通语句可以直接按〈Enter〉键运行并立刻输出结果，而选择结构、循环结构、函数定义、类定义、with 块等属于一条复合语句，需要按两次〈Enter〉键才能执行。

如果要执行大段代码，也为了方便反复修改，可以在 IDLE 中选择“File”→“New File”命令来创建一个程序文件，将其保存为扩展名为.py 或.pyw 的文件，然后按〈F5〉键或选择“Run”→“Run Module”命令运行程序，结果会显示到交互式窗口中，如图 1-3 所示。

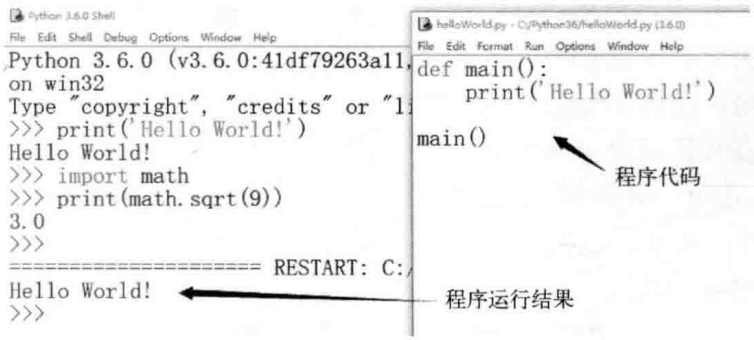


图 1-3 使用 IDLE 编写和运行 Python 程序

1.3.2 Anaconda3

Anaconda3 的安装包集成了大量常用的扩展库，并提供 Jupyter Notebook 和 Spyder 两个开发环境，得到了广大初学者和教学、科研人员的喜爱，是目前比较流行的 Python 开发环境之一。从官方网站 https://www.anaconda.com/download/下载合适版本并安装，然后启动 Jupyter Notebook 或 Spyder 即可。

（1）Jupyter Notebook

启动 Jupyter Notebook 会打开一个网页，在该网页右上角单击菜单“New”，然后选

择“Python 3”打开一个新窗口，即可编写和运行 Python 代码，如图 1-4 所示。另外，还可以选择“File”→“Download as”命令将当前代码以及运行结果保存为不同形式的文件，方便日后学习和演示，如图 1-5 所示。

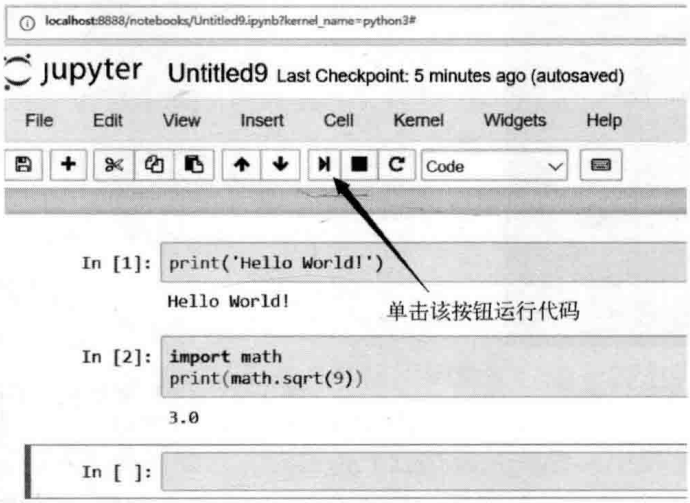


图 1-4　Jupyter Notebook 运行界面

图 1-5　保存 Jupyter Notebook 代码和运行结果

（2）Spyder

Anaconda3 自带的集成开发环境 Spyder 同时提供了交互式开发界面和程序编程与运

行界面，以及程序调试和项目管理功能，使用非常方便。在图 1-6 中，1 表示交互式运行，2 表示程序编写窗口，单击工具栏中绿色的“Fun File”按钮运行程序并在交互式窗口中显示运行结果，如图中 3 所示。

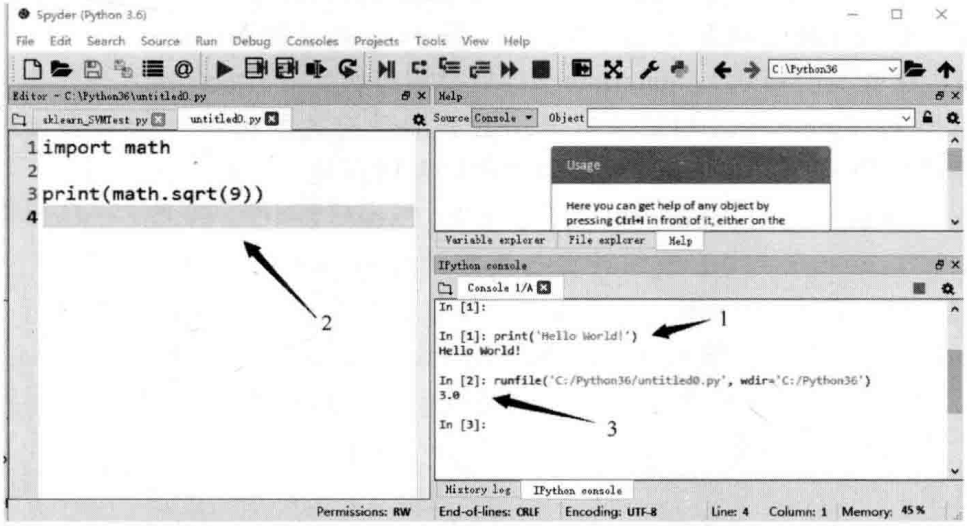


图 1-6　Spyder 运行界面

1.4　Python 编程规范

Python 非常重视代码的可读性，对代码布局和排版有更加严格的要求。这里重点介绍 Python 社区对代码编写的一些共同的要求、规范和一些常用的代码优化建议，最好在开始编写第一段代码的时候就要遵循这些规范和建议，养成一个好的习惯。

1）严格使用缩进来体现代码的逻辑从属关系。Python 对代码缩进是硬性要求的，这一点必须时刻注意。在函数定义、类定义、选择结构、循环结构、with 语句等结构中，对应的函数体或语句块都必须有相应的缩进，并且一般以 4 个空格为一个缩进单位。

2）每个 import 语句只导入一个模块，最好按标准库、扩展库、自定义库的顺序依次导入。尽量避免导入整个库，最好只导入确实需要使用的对象。

3）最好在每个类、函数定义和一段完整的功能代码之后增加一个空行，在运算符两侧各增加一个空格，逗号后面增加一个空格。

4）尽量不要写过长的语句。如果语句过长，可以考虑拆分成多个短一些的语句，以保证代码具有较好的可读性。如果语句确实太长而超过屏幕宽度，最好使用续行符“\”，或者使用圆括号把多行代码括起来表示是一条语句。

5）书写复杂的表达式时，建议在适当的位置加上括号，这样可以使得各种运算的隶属关系和顺序更加明确。

6）对关键代码和重要的业务逻辑代码进行必要的注释。在 Python 中有两种常用的注释形式：#和三引号。#用于单行注释，三引号常用于大段说明性文本的注释。

1.5 扩展库安装方法

在 Python 中，库或模块是指一个包含函数定义、类定义或常量的 Python 程序文件，一般并不对这两个概念进行严格区分。除了 math（数学模块）、random（与随机数以及随机化有关的模块）、datetime（日期时间模块）、collections（包含更多扩展性序列的模块）、functools（与函数以及函数式编程有关的模块）、tkinter（用于开发 GUI 程序的模块）、urllib（与网页内容读取以及网页地址解析有关的模块）等大量标准库之外，Python 还有 openpyxl（用于读写 Excel 文件）、python-docx（用于读写 Word 文件）、numpy（用于数组计算与矩阵计算）、scipy（用于科学计算）、pandas（用于数据分析）、matplotlib（用于数据可视化或科学计算可视化）、scrapy（爬虫框架）、shutil（用于系统运维）、pyopengl（用于计算机图形学编程）、pygame（用于游戏开发）、sklearn（用于机器学习）、tensorflow（用于深度学习）等几乎渗透到所有领域的扩展库或第三方库。到目前为止，Python 的扩展库已经超过 13 万个，并且每天还在增加。

在标准的 Python 安装包中，只包含了标准库，并不包含任何扩展库，开发人员根据实际需要再选择合适的扩展库进行安装和使用。Python 自带的 pip 工具是管理扩展库的主要方式，支持 Python 扩展库的安装、升级和卸载等操作。常用 pip 命令的使用方法如表 1-1 所示。

表 1-1　常用 pip 命令的使用方法

pip 命令示例	说　明
pip freeze[>requirements.txt]	列出已安装模块及其版本号
pip install SomePackage[==version]	在线安装 SomePackage 模块的指定版本
pip install SomePackage.whl	通过 whl 文件离线安装扩展库
pip install package1 package2 ...	依次（在线）安装 package1、package2 等扩展模块
pip install -r requirements.txt	安装 requirements.txt 文件中指定的扩展库
pip install --upgrade SomePackage	升级 SomePackage 模块
pip uninstall SomePackage[==version]	卸载 SomePackage 模块

有些扩展库安装时要求本机已安装相应版本的 C/C++编译器，或者有些扩展库暂时还没有与本机 Python 版本对应的官方版本，这时可以从http://www.lfd.uci.edu/～gohlke/pythonlibs/下载对应的.whl 文件（注意，一定不要修改文件名），然后在命令提示符环境中使用 pip 命令进行安装。例如：

```
pip install pygame-1.9.2a0-cp35-none-win_amd64.whl
```

注意，如果计算机上安装了多个版本的 Python 或者开发环境，最好切换至相应版本 Python 安装目录的 scripts 文件夹中，然后再在命令提示符环境中执行 pip 命令。如果要离线安装扩展库，也要把 whl 文件下载到相应的 scripts 文件夹中。