JetBrains大中华区市场部经理**赵磊作序**

300个实战案例 | 10万行源代码 | 22个综合实战项目

宁哥大讲堂

Python Crawler Programming

Principles, Technologies and Development

Python爬虫技术

深入理解原理、技术与开发

李宁◎编著

Li Ning

清華大學出版社

海量资源

宁哥大讲堂

Python Crawler Programming

Principles, Technologies and Development

Python爬虫技术

深入理解原理、技术与开发

李宁◎编著

Li Ning

清華大學出版社

北京

内容简介

本书从实战角度系统讲解 Python 爬虫的核心知识点，并通过大量的真实项目让读者熟练掌握 Python 爬虫技术。本书用 20 多个实战案例，完美演绎了使用各种技术编写 Python 爬虫的方式，读者可以任意组合这些技术，完成非常复杂的爬虫应用。

全书共 20 章，分为 5 篇。第 1 篇基础知识（第 1、2 章），主要包括 Python 运行环境的搭建、HTTP 基础、网页基础（HTML、CSS、JavaScript 等）、爬虫的基本原理、Session 与 Cookie。第 2 篇网络库（第 3 ～ 6 章），主要包括网络库 urllib、urllib3、requests 和 Twisted 的核心使用方法，如发送 HTTP 请求、处理超时、设置 HTTP 请求头、搭建和使用代理、解析链接、Robots 协议等。第 3 篇解析库（第 7 ～ 10 章），主要包括 3 个常用解析库（lxml、Beautiful Soup 和 pyquery）的使用方法，同时介绍多种用于分析 HTML 代码的技术，如正则表达式、XPath、CSS 选择器、方法选择器等。第 4 篇数据存储（第 11、12 章），主要包括 Python 中数据存储的解决方案，如文件存储和数据库存储，其中数据库存储包括多种数据库，如本地数据库 SQLite、网络数据库 MySQL 以及文档数据库 MongoDB。第 5 篇爬虫高级应用（第 13 ～ 20 章），主要包括 Python 爬虫的一些高级技术，如抓取异步数据、Selenium、Splash、抓取移动 App 数据、Appium、多线程爬虫、爬虫框架 Scrapy，最后给出一个综合的实战案例，综合了 Python 爬虫、数据存储、PyQt5、多线程、数据可视化、Web 等多种技术实现一个可视化爬虫。

本书可以作为广大计算机软件技术开发者、互联网技术研究人员学习“爬虫技术”的参考用书。也可以作为高等院校计算机科学与技术、软件工程、人工智能等专业的教学参考用书。

图书在版编目 (CIP) 数据

Python 爬虫技术：深入理解原理、技术与开发 / 李宁编著. —北京：清华大学出版社，2020.1（2020.5重印）
（宁哥大讲堂）
ISBN 978-7-302-53568-3

Ⅰ. ① P… Ⅱ. ①李… Ⅲ. ①软件工具—程序设计 Ⅳ. ① TP311.561

中国版本图书馆 CIP 数据核字（2019）第 173106 号

责任编辑： 盛东亮　钟志芳
封面设计： 李召霞
责任校对： 白　蕾
责任印制： 丛怀宇

出版发行： 清华大学出版社
网　　址： http://www.tup.com.cn，http://www.wqbook.com
地　　址： 北京清华大学学研大厦 A 座　　**邮　　编：** 100084
社 总 机： 010-62770175　　**邮　　购：** 010-62786544
投稿与读者服务： 010-62776969，c-service@tup.tsinghua.edu.cn
质 量 反 馈： 010-62772015，zhiliang@tup.tsinghua.edu.cn
印 装 者： 三河市铭诚印务有限公司
经　　销： 全国新华书店
开　　本： 203mm × 260mm　　**印　　张：** 31.25　　**字　　数：** 855 千字
版　　次： 2020 年 1 月第 1 版　　**印　　次：** 2020 年 5 月第 2 次印刷
定　　价： 89.00 元

产品编号：082638-01

PREFACE 前 言

Python 现在非常火爆。但 Python 就和英语一样，如果只会 Python 语言，就相当于只能用英语进行日常会话。然而，真正的英语高手是可以作为专业领域翻译的，如 IT、金融、数学等专业领域。Python 也是一样，光学习 Python 语言是不行的，要想找到更好的工作，或得到更高的薪水，需要学会用 Python 做某一领域的应用。

现在 Python 应用的热门领域比较广，例如人工智能，不过人工智能不光涉及 Python 语言本身的技术，还涉及数学领域的知识，虽然比较火爆，但绝对不是短时间可以掌握的。然后有一个领域与人工智能的火爆程度相当，但不像人工智能那样难入门，这就是爬虫领域。

为什么爬虫领域如此火爆呢？其实爬虫的基本功能就是从网上下载各种类型的数据（如 HTML、图像文件等）。但不要小瞧这些下载的数据，因为这些数据将成为很多应用的数据源。例如，著名的 Google 搜索引擎，每天都会有数以亿计的查询请求，而搜索引擎为这些请求返回的数据，都是来源于强大的爬虫。编写搜索引擎的第一步就是通过爬虫抓取整个互联网的数据，然后将这些数据库保存到本地（以特定的数据格式），接下来就是对这些数据进行分析整理。然后才可以通过搜索引擎进行查询。虽然搜索引擎的实现技术非常多，也非常复杂，但爬虫是 1，其他的所有技术都是 0，如果没有爬虫搜集数据，再强大的分析程序也毫无用武之地。

除了搜索引擎外，人工智能中的重要分支深度学习也需要爬虫抓取的数据来训练模型。例如，要想训练一个识别金字塔的深度学习模型，就需要大量与金字塔相关的图片进行训练。最简单的方式，就是使用百度或谷歌搜索金字塔图片，然后用爬虫抓取这些图片到本地。这是利用了搜索引擎通过关键字分类的特性，并且重新利用了这些分类的图片。

通过这些例子可以了解到，学习爬虫是进入其他更高端领域的钥匙，所以学习 Python 爬虫将成为第一个需要选择的热门领域。

尽管爬虫的基本功能是下载文件，但一个复杂的爬虫应用，可不光涉及网络技术。将数据下载后，还需要对数据进行分析，提取需要的信息，以及进行数据可视化，甚至需要一个基于 UI 的可视化爬虫。所以与爬虫有关的技术还是很多的。

由于 Pythonp 爬虫涉及的技术很多，学习资料过于分散。所以，笔者觉得很有必要编写一本全面介绍 Python 爬虫实战类的书籍，在书中分享笔者对 Python 爬虫以及相关技术的理解和经验，帮助同行和感兴趣的朋友快速入门，并利用 Python 语言编写各种复杂的爬虫应用。笔者希望本书能起到抛砖引玉的作用，使读者对 Python 爬虫以及相关技术产生浓厚的兴趣，并能成功进入 Python 爬虫领域。加油！高薪的工作在等着你们！

本书使用最新的 Python 3 编写，并在书中探讨了关于 Python 爬虫的核心技术。全书分 5 篇，共 20 章。内容涵盖 Python 爬虫的基础知识、常用网络库、常用分析库、数据存储技术、异步数据处理、可见即可爬技术、抓取移动 App、Scrapy 等。本书还包含 20 多个真实的项目，以便让读者身临其境

地体验 Python 爬虫的魅力。

限于篇幅，本书无法囊括 Python 爬虫以及相关技术的方方面面，只能尽自己所能，与大家分享尽可能多的知识和经验。相信通过本书的学习，读者可以拥有进一步深入学习的能力，达到 Python 爬虫高手的程度也只是时间问题。

为方便读者学习，本书配套提供了程序代码，并录制了一集视频，扫码即可下载程序代码或观看视频。

最后，笔者希望本书能为国内的 Python 爬虫以及相关技术的普及，为广大从业者提供有价值的实践经验并帮助他们快速上手贡献绵薄之力。

编著者

2019 年 10 月

CONTENTS 目 录

第 5 篇 爬虫高级应用

第1篇
基础知识

CHAPTER 1

第 1 章

开发环境配置

“工欲善其事，必先利其器。”，由于本书涉及的爬虫使用 Python 语言编写，所以在学习编写爬虫之前，必须先搭建好 Python 开发环境。Python 程序可以直接使用记事本开发，也可以使用 IDE（Integrated Development Environment，集成开发环境）开发。不过大多数项目都会使用 IDE 开发。因为 IDE 支持代码高亮、智能提示和可视化等功能，这些功能可以让开发效率大大提升。

本章主要介绍以下内容：

（1）安装 Python 标准环境；

（2）安装 Anaconda Python 环境；

（3）设置 PATH 环境变量；

（4）安装 PyCharm；

（5）配置 PyCharm。

关注公众号并输入 247415
下载配书源代码

1.1 安装官方的 Python 运行环境

不管用什么工具开发 Python 程序，都必须先安装 Python 的运行环境。由于 Python 是跨平台的，所以在安装之前，先要确定在哪一个操作系统平台上安装，目前最常用的是 Windows、macOS 和 Linux 三大平台。由于目前国内使用 Windows 和 macOS 的程序员比较多，所以本章主要以 Windows 和 macOS 为例介绍如何安装和使用 Python 运行环境。

读者可以直接到 Python 的官网（https://www.python.org/downloads）下载相应操作系统平台的 Python 安装包。

进入下载页面，浏览器会根据不同的操作系统显示不同的 Python 安装包下载链接。如果读者使用的是 Windows 平台，会显示如图 1-1 所示的 Python 下载页面。

如果读者使用的是 macOS 平台，会显示如图 1-2 所示的 Python 下载页面。

不管是哪个操作系统平台的下载页面，都会出现“Download Python 3.7.2”按钮（随着时间的推移，可能版本号略有不同）。直接单击“Download Python3.7.2”按钮下载相应平台的 Python 安装

包即可。如果是 Windows 平台，下载的是 exe 文件，如果是 macOS 平台，下载的是 pkg 文件，这是 macOS 上的安装程序，直接安装即可。

图 1-1　Windows 平台的 Python 下载页面

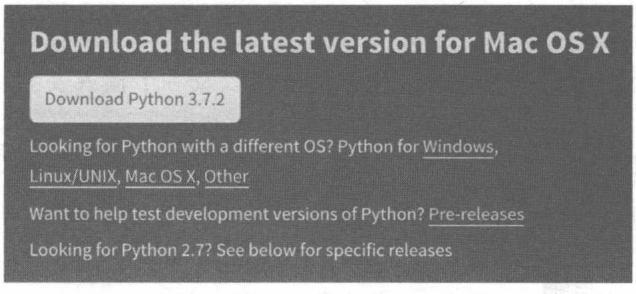

图 1-2　macOS 平台的 Python 下载页面

现在主要介绍在 Windows 平台如何安装 Python 运行环境。首先运行下载的 exe 文件，会显示 Python 安装界面。如果读者的机器已经安装了 Python 环境，那么会显示如图 1-3 所示的升级或重新安装的界面。如果读者的机器没有安装过 Python 环境，显示的安装界面与图 1-3 的界面类似，只是 Upgrade Now 变成了 Install Now。建议读者选中界面下方的“Add Python 3.7 to PATH”复选框，这样安装程序就会自动将 Python 的路径加到 PATH 环境变量中。

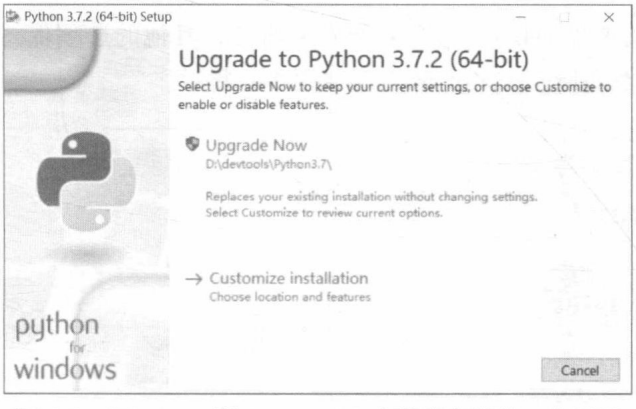

图 1-3　Windows 版 Python 3.7 安装程序的第一个界面

在图 1-3 所示的界面中出现两个安装选项，Upgrade Now 和 Customize installation，如果读者要升级 Python，可以单击 Upgrade Now 按钮，如果读者想定制安装选项，可以单击 Customize installation。现在我们定制安装 Python 环境。单击 Customize installation 按钮，会显示如图 1-4 所示的界面，默认所有的复选框全部选中，保留默认设置，然后单击 Next 按钮进入下一个安装界面。

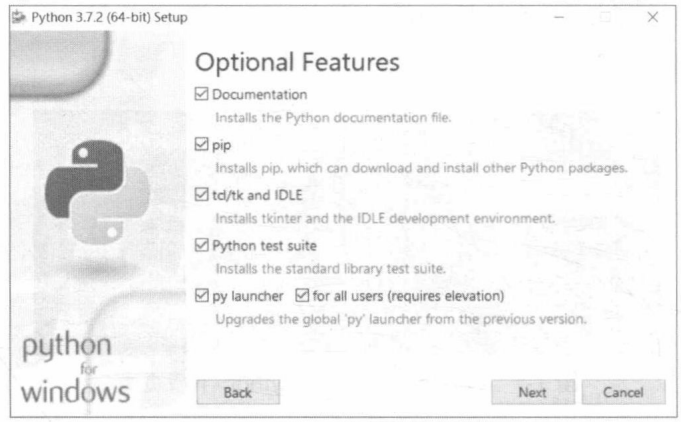

图 1-4 Python 选项界面

进入下一个安装界面后，会看到如图 1-5 所示的选项以及一个输入安装路径的文本框。

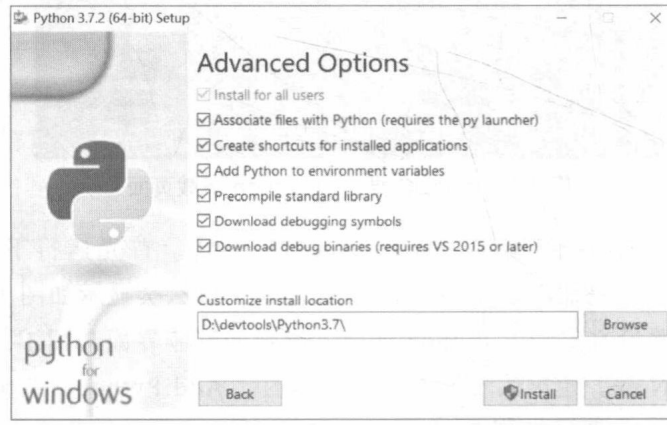

图 1-5 Python 安装程序的高级选项

读者可以在这个高级选项界面指定 Python 的安装路径，其他的选项保持默认设置即可。最后单击 Install 按钮安装 Python 开发环境，安装进度界面如图 1-6 所示。

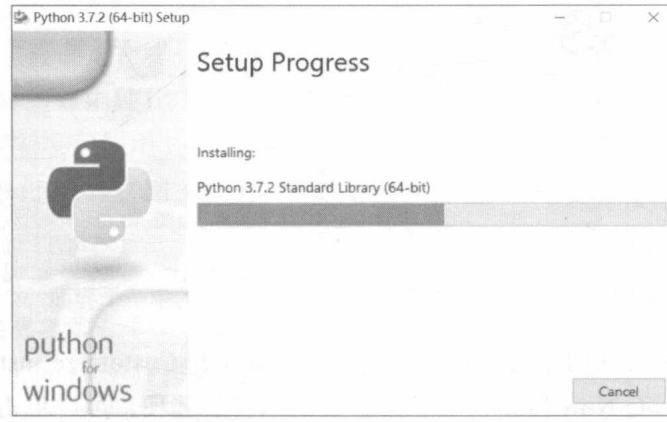

图 1-6 Python 安装进度界面

安装完成后，关闭安装界面。macOS 下安装 Python 的过程与 Windows 类似，读者可以自行安装。

1.2　配置 PATH 环境变量

在安装完 Python 运行环境后，我们可以测试一下 Python 运行环境，如果在安装 Python 的过程中忘记了选中“ Add Python 3.7 to PATH”复选框，那么默认情况下，Python 安装程序是不会将 Python 安装目录添加到 PATH 环境变量中的。这样一来，我们就无法在 Windows 命令行工具中的任何目录执行 python 命令了，必须进入 Python 的安装目录才可以使用 python 命令。

为了更方便地执行 python 命令，建议将 Python 安装目录添加到 PATH 环境变量中。在 Windows 平台配置 PATH 环境变量的步骤如下。

回到桌面，右击“此电脑”，在弹出菜单中单击“属性”菜单项，会显示如图 1-7 所示的“系统”窗口。

图 1-7　“系统”窗口

单击“系统”窗口左侧的“高级系统设置”，弹出如图 1-8 所示的“系统属性”窗口。

单击“系统属性”窗口下方的“环境变量 (N)...”按钮，弹出如图 1-9 所示的“环境变量”窗口。

图 1-8　“系统属性”窗口

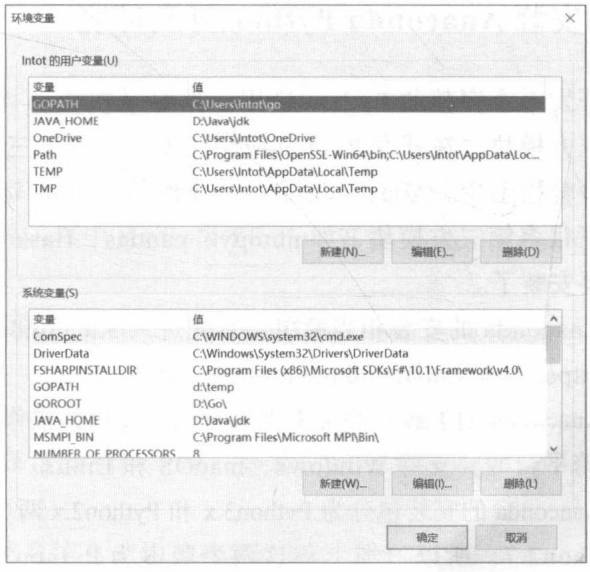

图 1-9　“环境变量”窗口

“环境变量”窗口有两个列表，上面的列表是为 Windows 当前登录用户设置环境变量，在这里设

置的环境变量只对当前登录用户有效。下面的列表是对所有用户设置的环境变量，这些变量对所有的用户都有效。读者在哪里设置 PATH 环境变量都可以，本书在上面的列表中设置了 PATH 环境变量。如果在列表中没有 PATH 环境变量，可单击“新建 (N)...”按钮添加一个新的 PATH 环境变量。如果已经有了 PATH 环境变量，双击 PATH，就会弹出如图 1-10 所示的“编辑环境变量”对话框。单击“新建”按钮，添加 Python 的安装路径即可。注意，这里要填写目录，而不是 python.exe 文件的路径。

在 Path 环境变量中添加 Python 的路径后，打开 Windows 命令行工具，执行 python --version 命令，如果输出如图 1-11 所示的内容，说明 Python 已经安装成功。

图 1-10 “编辑环境变量”对话框

图 1-11 测试 Python 运行环境是否安装成功

1.3 安装 Anaconda Python 开发环境

开发一个完整的 Python 应用，光使用 Python 本身提供的模块是远远不够的，因此，需要使用大量第三方模块。在开发 Python 应用时安装这些第三方模块是一件令人头痛的事，不过 Anaconda 会让这件事轻松不少。Anaconda 是一个集成的 Python 运行环境。除了包含 Python 本身的运行环境外，还集成了很多第三方模块，如 numpy、pandas、flask 等。也就是说，安装了 Anaconda 后，这些模块都不需要安装了。

Anaconda 的安装相当简单，首先进入 Anaconda 的下载页面，地址为：

https://www.anaconda.com/distribution

Anaconda 的下载页面会根据用户当前使用的操作系统自动切换到相应的 Anaconda 安装包。Anaconda 是跨平台的，支持 Windows、macOS 和 Linux。

Anaconda 的安装包分为 Python3.x 和 Python2.x 两个版本，目前 Anaconda 稳定版本分别支持 Python3.6 和 Python2.7，所以习惯上称这两个版本为 Python3.6 版和 Python2.7 版，尽管目前 Python 的最新版本是 3.7，但使用 Python3.6 仍然可以完美地运行本书的案例，所以读者只需要选择 Python3.6 或其以上版本即可。