

网页、接口、系统、软件、App  
自动化开发核心技术全解析

  
源代码下载

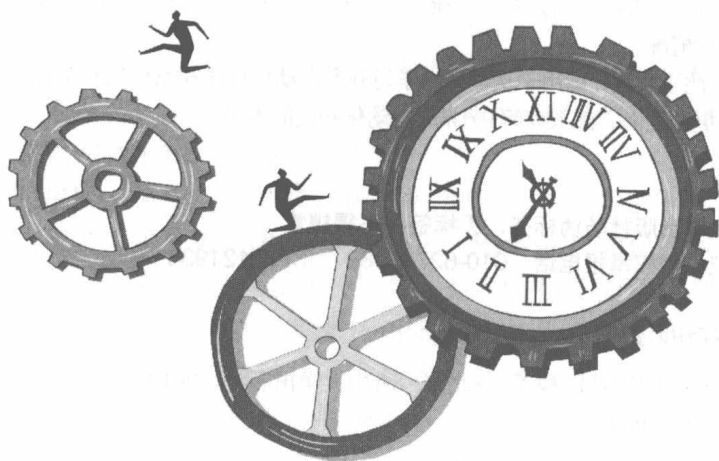


黄永祥 / 著

课外借



清华大学出版社



# Python

## 自动化开发实战

黄永祥 / 著

清华大学出版社  
北京

## 内 容 简 介

本书站在初学者的角度，从原理到实践，循序渐进地讲述了使用 Python 实现自动化开发的核心技术。全书从逻辑上可分为 Python 基础知识、Python 自动化技术和自动化系统开发三部分。Python 基础知识部分主要介绍 Python 的变量、数据类型、流程控制语句、函数与类等基础语法。Python 自动化技术部分分别介绍网页、计算机系统、软件和手机的自动化技术，并将自动化技术与人工智能的计算机视觉结合使用，使自动化程序更为稳定和智能。自动化系统开发部分介绍如何开发一个统一调度和管理自动化程序的 Web 系统，通过该系统可实现分布式管理自动化程序的运行情况。

本书由浅入深，注重实战，适用于从零开始学习开发自动化程序和系统的初学者，或者已经有一些自动化程序开发经验，但希望更加全面、深入理解 Python 自动化开发的人员。

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

### 图书在版编目 (CIP) 数据

Python 自动化开发实战/黄永祥著. —北京：清华大学出版社，2019  
ISBN 978-7-302-52490-8

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

中国版本图书馆 CIP 数据核字 (2019) 第 043089 号

责任编辑：王金柱

封面设计：王 翔

责任校对：闫秀华

责任印制：宋 林

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](mailto:c-service@tup.tsinghua.edu.cn)

质 量 反 馈：010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印 装 者：三河市龙大印装有限公司

经 销：全国新华书店

开 本：180mm×230mm

印 张：17.25

字 数：359 千字

版 次：2019 年 5 月第 1 版

印 次：2019 年 5 月第 1 次印刷

定 价：69.00 元

# 前 言

Python 一直是开发自动化程序的首选编程语言，随着大数据和人工智能的兴起，很多企业投身于智能化和自动化开发。Python 开发自动化程序不再仅限于自动化测试，它已应用于网络爬虫和业务流程自动化等方面，将重复性的业务交由程序处理，从而释放劳动力，正因如此，自动化开发成为当下最为追捧的技术之一。

本书站在初学者的角度，从原理到实践，循序渐进地讲述了使用 Python 进行自动化开发的核心技术。全书从逻辑上可分为 Python 基础知识、Python 自动化技术和开发自动化系统三部分。Python 基础知识主要介绍 Python 的变量、数据类型、流程控制语句、函数与类等基础语法，帮助不熟悉编程的读者快速掌握 Python 编程技巧。Python 自动化技术分别介绍网页、计算机系统、软件和手机的自动化开发技术，并将自动化开发与人工智能的计算机视觉结合使用，使自动化程序更为稳定和智能。自动化系统是将所有自动化程序统一调度和管理的 Web 系统，本部分通过开发一个自动化系统来实现分布式管理自动化程序的运行情况。

本书是笔者使用 Python 编写自动化程序和开发自动化系统的经验总结，全书循序渐进，由浅入深，结合当前各种热点新技术，从事软件自动化开发和编写自动化程序及进行软件自动化测试的读者能够从本书中获得收益。

## 本书结构

本书共分 15 章，从逻辑上可分为三部分：

第 1 部分，第 1~7 章讲述 Python 的基础知识，主要内容包括：搭建开发环境、变量与运算符、数据类型与控制语句、函数与类以及异常机制。

第 2 部分，第 8~13 章讲述 Python 的自动化技术，主要内容包括：网页自动化、接口自动化、系统自动化、软件自动化、利用计算机视觉实现自动化以及手机 App 自动化。

第 3 部分，第 14~15 章讲述自动化系统的开发，由 Python 的 Flask 框架实现，首先介绍 Flask 的基础知识，然后讲述自动化系统的开发过程。

## 本书特色

**循序渐进，知识全面：**本书站在初学者的角度，围绕 Python 的自动化技术展开讲解，从初学者必备基础知识着手，循序渐进地介绍了自动化程序开发和实现的各种知识，内容

难度适中，由浅入深，实用性强，覆盖面广，条理清晰，且具有较强的逻辑性和系统性。

**实例丰富，扩展性强：**本书每个知识点都单独以一个项目为例进行讲解，力求让读者更容易地掌握知识要点。本书实例经过作者的精心设计和挑选，根据编者的实际开发经验总结而来，涵盖在实际开发中遇到的各种问题。

**基于理论，注重实践：**在讲解的过程中，不仅介绍理论知识，而且安排了综合应用实例或小型应用程序，将理论应用到实践中，加强读者的实际开发能力，巩固开发技能和相关知识。

### 源代码下载

本书源代码的 github 下载地址：

<https://github.com/xyjw/python-Automation>

也可以扫描右侧二维码下载。

如果你在下载过程中遇到问题，可发送邮件至 [554301449@qq.com](mailto:554301449@qq.com) 获得帮助，邮件标题为“Python 自动化开发实战下载资源”。



### 技术服务

读者在学习或者工作的过程中，如果遇到实际问题，可以加入 QQ 群 93314951 与笔者联系，笔者会在第一时间给予回复。

### 读者对象

本书主要适合以下读者阅读：

- ◆ 从零开始学习编写自动化程序的初学者和大学生
- ◆ Python 自动化开发工程师。
- ◆ 从事自动化测试和运维的技术人员。
- ◆ 培训机构及网课教学用书。

虽然笔者力求本书更臻完美，但由于水平所限，难免会出现错误，欢迎广大读者和高手专家给予指正，笔者将十分感谢。

编者

2019年3月

# 目 录

第 1 章 认识 Python .....	1
1.1 了解 Python .....	1
1.2 安装 Python 3 .....	2
1.3 安装 PyCharm .....	4
1.4 安装第三方模块 .....	7
1.5 我的“Hello World” .....	9
1.6 本章小结 .....	11
第 2 章 变量与运算符 .....	12
2.1 变量的命名与使用 .....	12
2.2 变量的深浅拷贝 .....	15
2.3 运算符的使用 .....	16
2.3.1 算术运算符 .....	17
2.3.2 比较运算符 .....	18
2.3.3 赋值运算符 .....	19
2.3.4 逻辑运算符 .....	20
2.3.5 位运算符 .....	21
2.3.6 成员运算符 .....	22
2.3.7 身份运算符 .....	23
2.3.8 运算符的优先级 .....	24
2.4 本章小结 .....	25
第 3 章 数据类型 .....	26
3.1 数字的类型及转换 .....	26
3.2 字符串的定义及使用 .....	28
3.2.1 字符串的定义 .....	28
3.2.2 字符串截取 .....	29
3.2.3 字符串替换 .....	30
3.2.4 字符串查找元素 .....	30
3.2.5 字符串分割 .....	32
3.2.6 字符串拼接 .....	32
3.3 元组与列表 .....	32
3.4 集合与字典 .....	37

3.5	数据类型的转化	40
3.5.1	字符串和列表的转换	40
3.5.2	字符串与字典的转换	41
3.5.3	列表与字典的转换	42
3.6	本章小结	43
<b>第4章</b>	<b>流程控制语句</b>	<b>45</b>
4.1	if 语句	45
4.2	for 循环	47
4.3	while 循环	50
4.4	推导式	52
4.5	三目运算符	54
4.6	实战：编写“猜数字”游戏	55
4.7	本章小结	56
<b>第5章</b>	<b>函数</b>	<b>58</b>
5.1	函数的定义	58
5.2	函数参数	59
5.3	函数的返回值	62
5.4	函数的调用	63
5.5	变量的作用域	64
5.6	实战：编写“猜词语”游戏	66
5.7	本章小结	68
<b>第6章</b>	<b>类与对象</b>	<b>69</b>
6.1	类的使用	69
6.2	类的封装	73
6.3	类的继承	74
6.4	实战：编写“过家家”游戏	76
6.5	本章小结	79
<b>第7章</b>	<b>异常机制</b>	<b>80</b>
7.1	了解异常	80
7.2	捕捉异常	82
7.3	自定义异常	84
7.4	实战：编写“角色扮演”游戏	85
7.5	本章小结	86



第 8 章 网页自动化开发	88
8.1 了解 Selenium	88
8.2 安装 Selenium	89
8.3 浏览器查找元素	93
8.4 Selenium 定位元素	97
8.5 Selenium 操控元素	101
8.6 Selenium 常用功能	105
8.7 实战：编写“百度自动答题”程序	113
8.8 本章小结	120
第 9 章 接口自动化开发	121
9.1 分析网站接口	121
9.2 Requests 概述及安装	124
9.3 简单的请求方式	125
9.4 复杂的请求方式	127
9.5 文件下载与上传	130
9.6 实战：编写“12306 车次查询”程序	132
9.7 本章小结	137
第 10 章 系统自动化开发	138
10.1 PyAutoGUI 概述及安装	138
10.2 截图与识别	139
10.3 鼠标控制功能	143
10.4 键盘控制功能	145
10.5 消息框功能	146
10.6 实战：编写“百度用户登录”程序	147
10.7 本章小结	151
第 11 章 软件自动化开发	153
11.1 PyWinAuto 概述及安装	153
11.2 查找软件信息	154
11.3 连接 CS 软件	157
11.4 基于 Uia 软件操控	161
11.5 基于 Win32 软件操控	166
11.6 从源码剖析 PyWinAuto	170
11.7 实战：自动撰写新闻稿	171
11.8 本章小结	179



第 12 章	图像识别与定位	181
12.1	OpenCV 概述及安装	181
12.2	图像特征点检测算法	183
12.3	图像匹配与定位	187
12.4	实战：自动打印 PDF 文件	193
12.5	本章小结	199
第 13 章	App 自动化开发	201
13.1	Appium 简介及原理	201
13.2	搭建开发环境	202
13.3	连接 Android 系统	209
13.4	定位元素	214
13.5	操控元素	216
13.6	实战：淘宝商品采集	220
13.7	本章小结	226
第 14 章	Flask 入门基础	228
14.1	概述与安装	228
14.2	快速实现一个简单的网站系统	229
14.3	路由编写规则	231
14.4	请求参数	233
14.5	响应过程	234
14.6	本章小结	236
第 15 章	自动化系统的开发与部署	238
15.1	系统设计概述	238
15.2	搭建开发环境	241
15.3	任务调度系统	244
15.3.1	配置文件	245
15.3.2	数据模型	246
15.3.3	Admin 后台	250
15.3.4	系统接口与运行	253
15.4	任务执行系统	255
15.4.1	配置文件	256
15.4.2	异步任务	257
15.4.3	系统接口与运行	259
15.5	系统上线部署	261
15.6	本章小结	267

# 第 1 章

---

## 认识 Python

本章主要介绍 Python 的概念、搭建 Python 的开发环境及实现简单的功能输出，通过本章的学习，使读者对 Python 有一定的了解和认知，并做好开发 Python 程序的准备。

### 1.1 了解 Python

---

Python 是一种面向对象的解释型计算机程序设计语言，由荷兰人 Guido van Rossum 于 1989 年发明，第一个公开发行人版发行于 1991 年。它是纯粹的自由软件，源代码和解释器 CPython 遵循 GPL（General Public License）协议，同时被称为胶水语言，能够把其他开发语言制作的各种模块（尤其是 C/C++）很轻松地联结在一起使用。

Python 为我们提供了非常完善的标准模块，覆盖了网络、文件、GUI、数据库和文本等大量功能模块，形象地称作“内置电池（batteries included）”。使用 Python 开发程序，许多功能不必从零编写，直接调用现成的即可。除了内置的模块外，Python 还有大量的第三方模块，这是别人开发的，并且免费供我们直接使用。如果我们开发的代码通过封装处理，也可以作为第三方模块给别人使用。

发明者 Guido van Rossum 给 Python 的定位是“优雅”、“明确”、“简单”，所以 Python 开发的程序看上去总是简单易懂，同一个功能，Python 的代码量比其他开发语言更为简洁。

初学者学习 Python，不但入门容易，而且将来深入下去，也可以编写非常复杂的程序。

任何编程语言都有缺点，Python 也不例外。Python 的主要缺点是运行速度慢，与 C 程序相比会显得非常慢，因为 Python 是解释型语言，代码在执行时会一行一行地翻译成 CPU 能理解的机器码，这个翻译过程非常耗时，而 C 程序是运行前直接编译成 CPU 能执行的机器码。

但是现在大量的应用程序不需要这么快的运行速度，因为用户根本感觉不出来。例如一个下载 MP3 的网络应用程序，C 程序的运行时间需要 0.001 秒，而 Python 程序的运行时间需要 0.1 秒，慢了 100 倍，但由于网络数据传输更慢，需要等待 1 秒，因此用户是无法感受到程序运行的速度。

因为 Python 是解释型语言，其代码是由 Python 解释器执行。整个 Python 语言从规范到解释器都是开源的，在理论上，只要水平够高，任何人都可以编写 Python 解释器来执行 Python 代码。目前，Python 的解释器主要有以下几种，如表 1-1 所示。

表 1-1 Python 主要的解释器

解释器	说明
CPython	Python 官方使用的解释器，由 C 语言开发，也是目前使用最广的 Python 解释器
IPython	IPython 是基于 CPython 之上的一个交互式解释器，在交互方式上有所增强，但是执行代码的功能和 CPython 是完全一样的
PyPy	PyPy 是另一个 Python 解释器，它的目标是执行速度。PyPy 采用 JIT 技术，对 Python 代码进行动态编译，因此提高了 Python 代码的执行速度
Jython	Jython 是运行在 Java 平台上的 Python 解释器，可以直接把 Python 代码编译成 Java 字节码执行
IronPython	IronPython 和 Jython 类似，只不过 IronPython 是运行在微软 .Net 平台上的 Python 解释器，可以直接把 Python 代码编译成 .Net 的字节码

## 1.2 安装 Python 3

目前，Python 主要分为两大版本：Python 2.X 和 Python 3.X。Python 核心团队计划在 2020 年停止支持 Python 2.X，现在很多第三方模块已开始不再支持 Python 2.X 的使用。因此本书以最新版本 Python 3.7 为例，讲述如何在 Windows 下安装 Python。

首先在浏览器上输入 <https://www.python.org/downloads/release/python-370/>，这是 Python 安装包的下载界面。在下载界面上找到 exe 安装包的下载地址并单击下载，安装包的下载地址如图 1-1 所示。

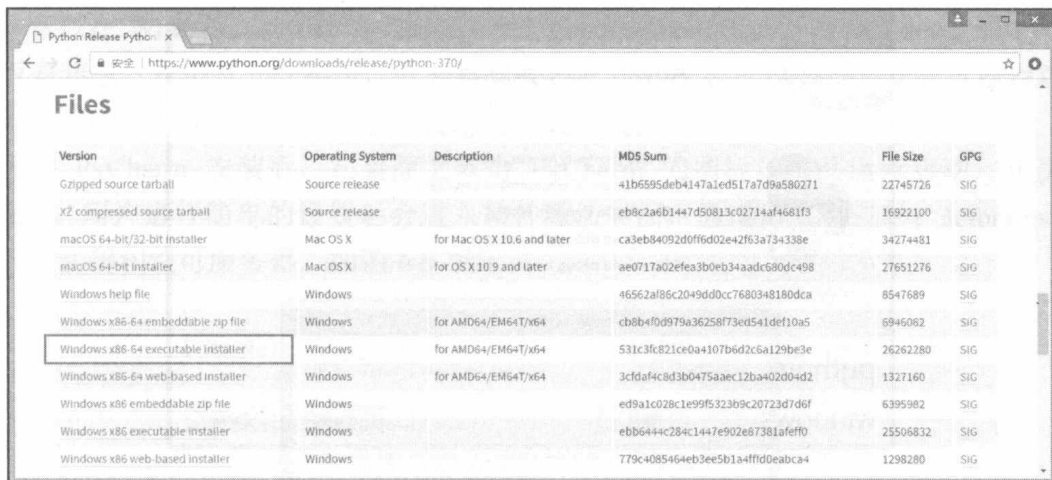


图 1-1 Python 下载地址

安装包下载完成后，在安装包所在路径打开 exe 安装包，然后勾选“Add Python 3.7 to Path”并单击“Customize installation”，如图 1-2 所示。

单击“Customize installation”将会进入“Optional Features”界面，在该界面上把全部选项勾选并单击“Next”按钮，如图 1-3 所示。

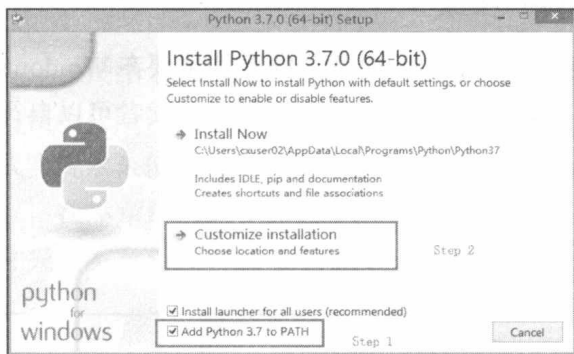


图 1-2 Python 安装界面

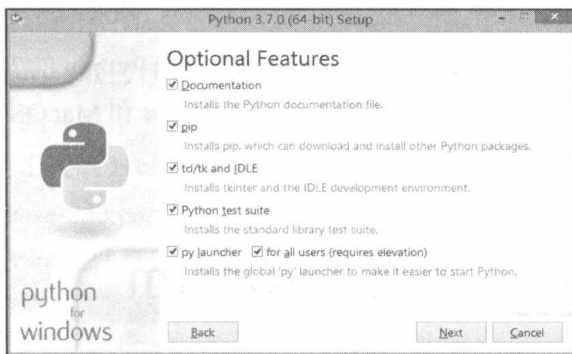


图 1-3 Optional Features 界面

从“Optional Features”界面进入到“Advanced Options”界面，将界面上的全部选项勾选后并设置 Python 的安装路径，本书将 Python 安装在 E:\Python，如图 1-4 所示。

最后单击 Install 按钮，等待程序完成安装即可。安装时间由于各个电脑与网络的差异会造成不同，只需耐心等待即可。安装完成后，打开 Windows 的命令符窗口，输入“python”并按回车键即可进入 Python 的交互模式，如图 1-5 所示。

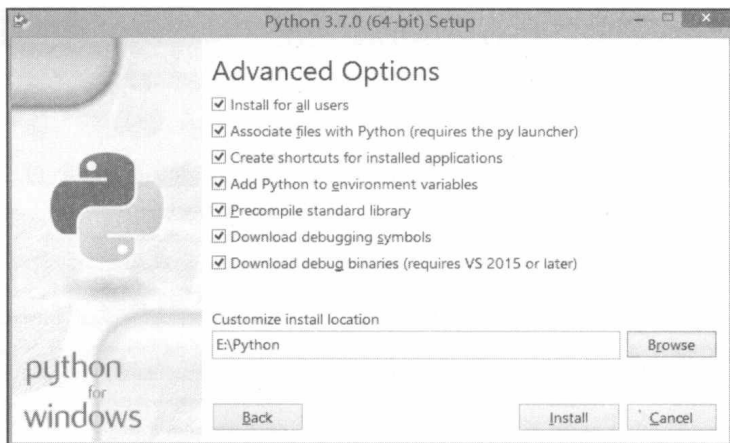


图 1-4 Advanced Options 界面

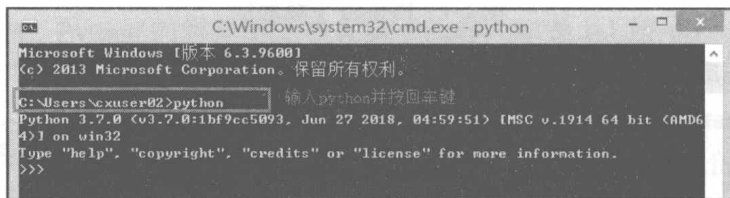


图 1-5 Python 的交互模式

至此，在 Windows 下已完成 Python 的安装。由于本书后面章节的实例主要在 Windows 下实现，所以本书不再讲述 Linux 和 MacOS 下安装 Python 的过程，有需要的读者可以自行网上搜索相关资料。

## 1.3 安装 PyCharm

PyCharm 是一种 Python IDE，它带有一整套可以帮助用户在使用 Python 语言开发时提高其效率的工具，比如调试、语法高亮、Project 管理、代码跳转、智能提示、自动完成、单元测试、版本控制等。此外，该 IDE 提供了一些高级功能，例如支持 Django 框架下的专业 Web 开发。

在浏览器输入下载地址 <http://www.jetbrains.com/pycharm/download>，可以看到 PyCharm 分别支持 Windows、Linux 和 MacOS 三大系统的使用，版本分为专业版和社区版。专业版提供了完整的功能，如专业的 Web 开发，但需要支付相应的费用；社区版完全免费使用，但只提供一些基础的开发功能。

本章以在 Windows 下安装 PyCharm 专业版为例，在官网下载 Windows 的 PyCharm 专业版安装包，双击打开安装包，并根据安装提示完成安装过程即可。安装过程相对较为简单，本书就不做详细的介绍。

完成 PyCharm 安装后，在桌面上双击 PyCharm 的图标，将其运行启动。初次运行 PyCharm，用户进行简单的设置后会进入软件激活界面，激活方式有三种：Jetbrains 用户激活、激活码和许可服务器。如图 1-6 所示。

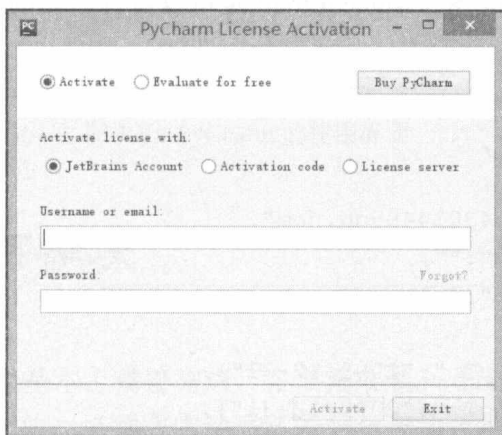


图 1-6 PyCharm 激活界面

在网上可以通过搜索“PyCharm 破解”分别获取激活码和许可服务器来激活 PyCharm 的使用权限，但这种方式有一个时效性，时间一到又要重新激活。也可以通过安装补丁的方式实现永久使用（此处仅作教学使用，建议使用正式版），具体操作步骤如下。

在本章的源码中找到 JetbrainsCrack-2.8-release-enc.jar 文件，并存放在本地系统某个文件夹，本章将文件存放在 PyCharm 的安装目录 bin 文件夹，如图 1-7 所示。

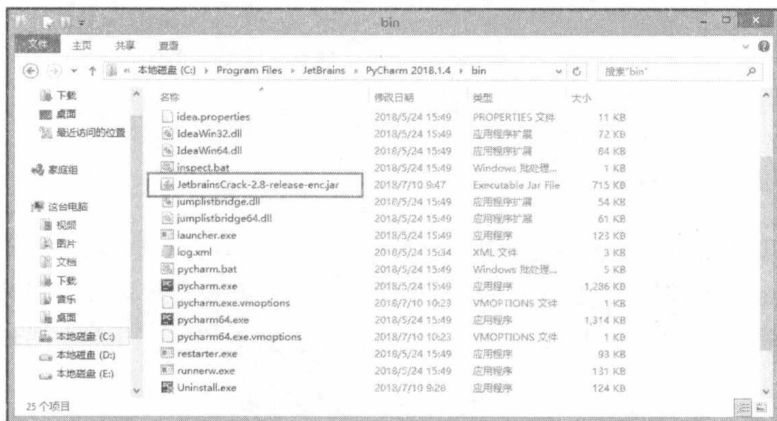


图 1-7 bin 文件夹信息

在 bin 文件夹下找到 `pycharm.exe.vmoptions` 和 `pycharm64.exe.vmoptions` 文件，并将两个文件以记事本的方式打开，分别在最下方添加相应代码并保存文件，添加代码如下：

```
-javaagent:C:/Program Files/JetBrains/PyCharm 2018.1.4/bin/  
JetbrainsCrack-2.8-release-enc.jar
```

最后运行 PyCharm，在激活界面选择 `Activation code` 并输入以下激活码激活，其中 `licenseeName`、`assigneeName` 和 `assigneeEmail` 是可以自行命名的。激活码如下所示：

```
ThisCrackLicenseId-{  
  "licenseId":"11011",  
  "licenseeName":"aa",  
  "assigneeName":"aa",  
  "assigneeEmail":"554301449@qq.com",  
  "licenseRestriction": "",  
  "checkConcurrentUse":false,  
  "products": [  
    {"code":"II", "paidUpTo":"2099-12-31"},  
    {"code":"DM", "paidUpTo":"2099-12-31"},  
    {"code":"AC", "paidUpTo":"2099-12-31"},  
    {"code":"RS0", "paidUpTo":"2099-12-31"},  
    {"code":"WS", "paidUpTo":"2099-12-31"},  
    {"code":"DPN", "paidUpTo":"2099-12-31"},  
    {"code":"RC", "paidUpTo":"2099-12-31"},  
    {"code":"PS", "paidUpTo":"2099-12-31"},  
    {"code":"DC", "paidUpTo":"2099-12-31"},  
    {"code":"RM", "paidUpTo":"2099-12-31"},  
    {"code":"CL", "paidUpTo":"2099-12-31"},  
    {"code":"PC", "paidUpTo":"2099-12-31"}  
  ],  
  "hash":"2911276/0",  
  "gracePeriodDays":7,  
  "autoProlongated":false}
```

PyCharm 成功激活后，将会进入 PyCharm 的使用界面，如图 1-8 所示。

至此，PyCharm 的安装和激活过程已讲述完成，上述的教程可能会随着 PyCharm 版本的更新而发生细微的差异。这里通过讲述 PyCharm 的安装和激活，让初学者对 PyCharm 有初步的认识。



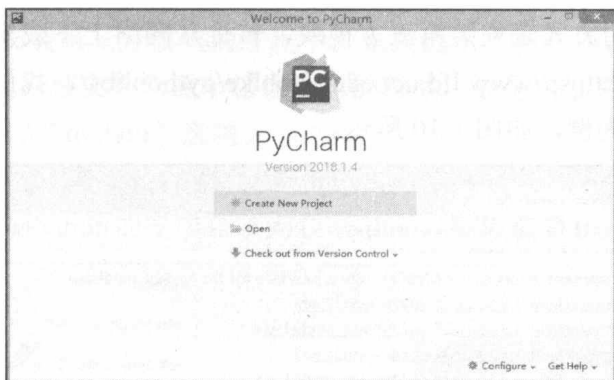


图 1-8 PyCharm 的使用界面

## 1.4 安装第三方模块

到此 Python 的开发环境基本上搭建完成，主要是安装 Python 和 PyCharm。在开发过程中，如果需要使用第三方模块，必须在本地系统安装该模块才能在代码中使用，否则代码在运行过程中会提示错误信息。安装第三方模块可以使用 Pip 执行安装，Pip 的安装方式有两种：在线安装和本地安装，具体安装过程说明如下：

使用 Pip 主要在 Windows 的命令提示符窗口输入安装指令即可完成，如果是其他系统可以在系统的终端输入安装指令。我们在 Windows 下打开命令提示符窗口，以安装第三方模块 requests 为例，在窗口中直接输入“pip install requests”指令，pip install 是固定的安装指令；requests 是模块名。我们只需等待指令执行完毕即可完成模块的安装，如图 1-9 所示。

```
C:\Windows\system32\cmd.exe
C:\Users\N.xuser@2>pip install requests
Collecting requests
  Using cached https://files.pythonhosted.org/packages/65/47/7e02164a2a3db58ed6d
8a6a81de6b0b69e4c3fd57e284257925dfc12ba/requests-2.19.1-py2.py3-none-any.whl
Collecting idna<2.8,>=2.5 (from requests)
  Using cached https://files.pythonhosted.org/packages/4b/2a/8276479a4b3cae8a8c
1af2f8e4355746a97fab05a372e442c6a6b876165/idna-2.7-py2.py3-none-any.whl
Collecting urllib3<1.24,>=1.21.1 (from requests)
  Using cached https://files.pythonhosted.org/packages/bd/c9/6f/d990019071a4a32a
5e7eb78a1d92e53851ef4f56f62a34866a7d8ffb/urllib3-1.23-py2.py3-none-any.whl
Collecting certifi>=2017.4.17 (from requests)
  Using cached https://files.pythonhosted.org/packages/7c/66/92ad559b7192d846975
fc916b65f667c798c3a32bea732349ffe9a5f5a5/certifi-2018.4.16-py2.py3-none-any.whl
Collecting chardet<3.1.0,>=3.0.2 (from requests)
  Using cached https://files.pythonhosted.org/packages/bc/a9/01ffe9bf15624274b4
37b4401d6ec7c455cc7510b2264c51f499894387/chardet-3.0.4-py2.py3-none-any.whl
Installing collected packages: idna, urllib3, certifi, chardet, requests
  You may wish to allow automatic updates by creating a file in your pip
  installation directory named 'pip-autoupdate.txt'.
Successfully installed certifi-2018.4.16 chardet-3.0.4 idna-2.7 requests-2.19.1
urllib3-1.23
C:\Users\N.xuser@2>
```

图 1-9 Pip 在线安装

如果以本地安装的方式来安装第三方模块，首先从网络上下载安装包，在浏览器中输入安装包的下载地址 <https://www.lfd.uci.edu/~gohlke/pythonlibs/>，找到与系统相对应的模块信息，以 Mysqlclient 为例，如图 1-10 所示。

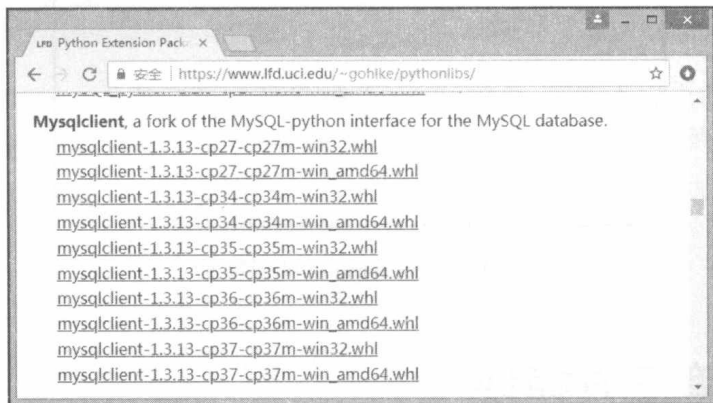


图 1-10 模块信息

以图 1-10 的 `mysqlclient-1.3.13-cp37-cp37m-win_amd64(win32).whl` 为例，该模块名包含以下信息：

- (1) `mysqlclient`: 模块名。
- (2) `1.3.13`: 模块的版本型号。
- (3) `cp37-cp37m`: 支持 Python 3.7 版本适用。
- (4) `win_amd64`: 支持 Windows 64 位系统使用。
- (5) `win32`: 支持 Windows 32 位系统使用。

模块下载后，在命令提示符使用 Pip 安装模块安装包，我们将安装包放置在本地 D 盘的根目录，然后输入指令“`pip install D:\mysqlclient-1.3.13-cp37-cp37m-win_amd64.whl`”即可完成安装，如图 1-11 所示。

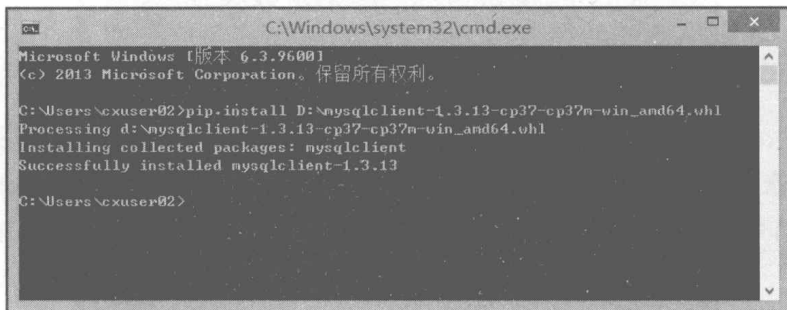


图 1-11 Pip 本地安装