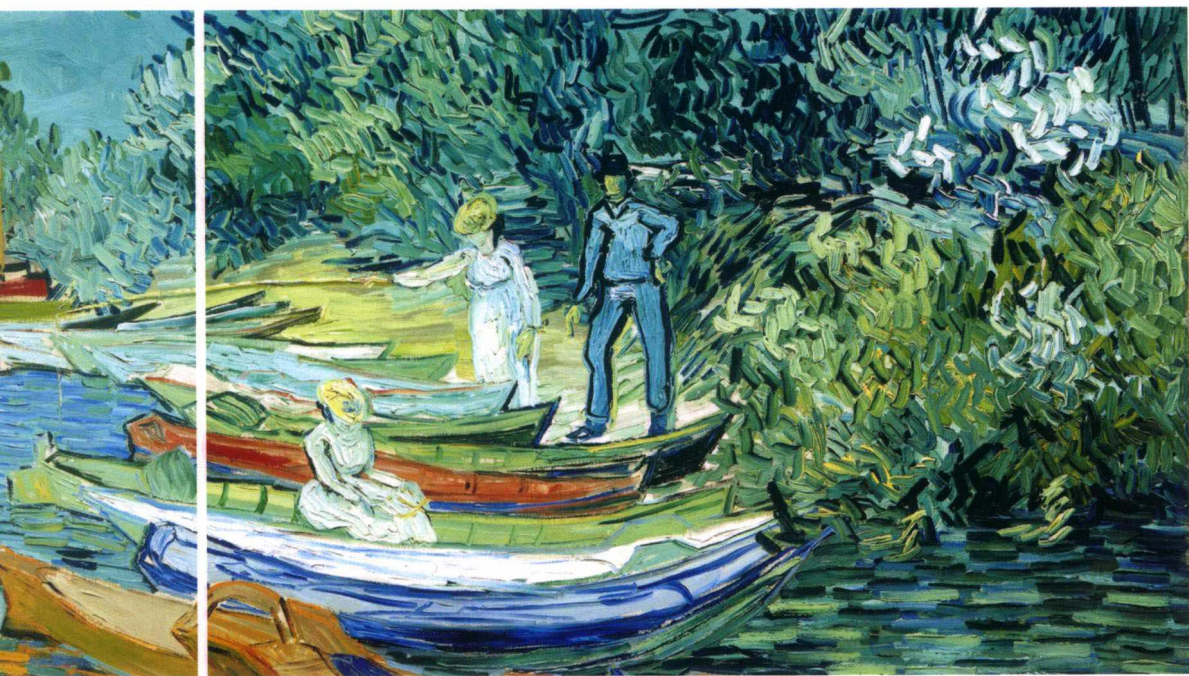




21世纪高等学校计算机
基础实用规划教材

二级Python编程指南

◎ 娄岩 主编



清华大学出版社





21世纪高等学校计算机
基础实用规划教材

二级Python编程指南

◎ 娄岩 主编 张志常 曹鹏 副主编



清华大学出版社
北京

内 容 简 介

本书以实用为导向,从基础概念、语法结构、应用实例、开发技巧等方面深入浅出地介绍了 Python 的程序设计方法以及项目实现流程。全书共 13 章,分别是 Python 概述、基础语法、控制结构、函数、模块与文件、面向对象程序设计、图形界面设计、数据库应用、网络及多线程、网络爬虫、图像操作与处理、科学计算与数据可视化应用、数据挖掘与机器学习和 Python 解析 XML。

本书以实例方式讲解,每个实例都通过验证、图文并茂、操作步骤完善并附有具体脚本代码,易于掌握和学习。本书提供开放式的课程网站 (<http://www.cmu.edu.cn/computer>) 和相应的课件作为支持。

本书既可以作为全国普通高校各专业计算机公共基础课程教材,又可作为全国计算机等级考试的辅导教材,还可作为专业技术人员的参考用书。

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

二级 Python 编程指南/娄岩主编. —北京:清华大学出版社,2019

(21 世纪高等学校计算机基础实用规划教材)

ISBN 978-7-302-52628-5

I. ①二… II. ①娄… 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 043402 号

责任编辑:贾 斌

封面设计:刘 键

责任校对:梁 毅

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

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京嘉实印刷有限公司

经 销 者:全国新华书店

开 本:185mm×260mm 印 张:13.5 字 数:328 千字

版 次:2019 年 4 月第 1 版 印 次:2019 年 4 月第 1 次印刷

印 数:1~2500

定 价:45.00 元

产品编号:081379-01

前 言

近年来, Python 已经快速发展成为最热门的语言之一, Python 在系统运维、Web 开发、网络爬虫、科学计算、图像处理、人工智能等领域都已得到了广泛的应用, 这大大促进了 Python 语言的发展。为适应新时代信息技术的发展, 教育部考试中心决定自 2018 年 3 月起, 在计算机二级考试中加入“Python 语言程序设计”科目。目前, 已经有越来越多的人开始投入到 Python 的应用研究中。

为此, 我们精心策划和编写了《二级 Python 编程指南》这本面向实践、注重应用的教材, 目的是使读者既能够结合实例学习 Python 程序设计方法, 提高运用 Python 编程并解决实际应用问题的能力; 又可以通过学习, 为参加全国计算机等级考试做好准备。

本书以实例方式进行讲解, 其中每个实例都通过验证、操作步骤完善并附有具体脚本代码。本书图文并茂, 知识难易程度循序渐进, 具有很强的可读性和可操作性。读者在学习过程中可以按图索骥, 在较短时间内掌握 Python 的开发技术。本书兼顾了不同专业、不同层次读者的需要, 以提高读者自主学习和运用知识的能力为目标, 强化了学习过程中实践能力的培养, 为 Python 初学者提供全面、翔实的参考资料, 使其易于掌握 Python 程序设计方法、项目开发流程和步骤。

全书共有 13 章, 分别是 Python 概述; 基础语法; 控制结构; 函数、模块与文件; 面向对象程序设计; 图形界面设计; 数据库应用; 网络及多线程; 网络爬虫; 图像操作与处理; 科学计算与数据可视化应用; 数据挖掘与机器学习; Python 解析 XML。第 1 章由娄岩编写, 第 2 章由郭美娜编写, 第 3 章由霍妍编写, 第 4 章由马瑾编写, 第 5 章由曹阳编写, 第 6 章由郭婷婷编写, 第 7 章由徐东雨编写, 第 8 章由刘佳编写, 第 9 章由曹鹏编写, 第 10 章由郑琳琳编写, 第 11 章由庞东兴编写, 第 12 章由张志常编写, 第 13 章由王艳华编写。

感谢出版社的编辑, 为本书的出版进行了精心策划、充分论证。在此向所有参加编写的同事们及帮助和指导过我们工作的朋友们表示衷心的感谢! 由于编者水平有限, 书中难免存在疏漏之处, 恳请广大读者批评斧正。

娄 岩
2018 年 6 月

目 录

第 1 章 Python 概述	1
1.1 Python 简介	1
1.1.1 Python 的概念	1
1.1.2 Python 的发展历史	1
1.1.3 Python 3	2
1.1.4 Python 的特点	2
1.1.5 Python 的应用领域	2
1.2 Python 的安装	3
1.3 第一个 Python 程序	5
1.4 搭建 Python 集成开发环境	6
1.5 Python 的输入/输出	12
1.5.1 Python 的输入	12
1.5.2 Python 的输出	13
本章小结	14
第 2 章 基础语法	15
2.1 Python 的基本数据类型	15
2.1.1 变量的赋值和数据类型	15
2.1.2 数字	16
2.1.3 字符串	16
2.1.4 列表	18
2.1.5 元组	19
2.1.6 字典	19
2.1.7 集合	20
2.2 Python 的基本运算符	21
2.2.1 算术运算符	21
2.2.2 字符串运算符	22
2.2.3 比较运算符	22
2.2.4 赋值运算符	23
2.2.5 逻辑运算符	23
2.2.6 成员运算符	24

2.2.7 运算符的优先级	24
本章小结	25
第3章 控制结构	26
3.1 顺序结构	26
3.2 分支结构	27
3.2.1 单分支结构 (if 语句)	27
3.2.2 双分支结构 (if...else 语句)	28
3.2.3 多分支结构 (if...elif...else 语句)	28
3.2.4 pass 语句	30
3.2.5 try...except 语句	30
3.3 循环结构	31
3.3.1 while 语句	31
3.3.2 for 语句	32
3.3.3 循环嵌套	32
3.3.4 break 语句和 continue 语句	33
本章小结	34
第4章 函数、模块与文件	35
4.1 函数	35
4.1.1 系统函数	35
4.1.2 函数的定义	38
4.1.3 函数的参数和返回值	38
4.1.4 变量的作用域	39
4.1.5 递归调用	39
4.2 模块	40
4.3 文件	41
4.3.1 文件的打开与关闭	41
4.3.2 文件的读操作	42
4.3.3 文件的写操作	43
4.3.4 文件的指针定位	45
本章小结	46
第5章 面向对象程序设计	47
5.1 面向对象程序设计基础	47
5.1.1 面向对象程序设计的基本概念	47
5.1.2 面向对象程序设计的基本特性	48
5.2 类与对象	49
5.2.1 定义类和对象	49

5.2.2	构造函数	50
5.2.3	实例属性和类属性	51
5.3	类的继承与多态	52
5.3.1	类的继承	52
5.3.2	类的多继承	53
5.3.3	类的多态	54
	本章小结	55
第 6 章	图形界面设计	56
6.1	Python 图形开发库	56
6.1.1	开发平台	56
6.1.2	创建 Windows 窗口	57
6.2	Tkinter 常用组件	58
6.2.1	Tkinter 组件	58
6.2.2	标签	60
6.2.3	按钮	62
6.2.4	文本框	64
6.2.5	列表框	65
6.2.6	单选按钮和复选框	66
6.2.7	消息窗口	69
6.2.8	对话框	70
6.2.9	菜单	71
6.3	窗口内图形绘制	73
6.3.1	Canvas 画布组件	73
6.3.2	绘制图形对象	73
6.4	Python 事件处理	76
6.4.1	事件类型	76
6.4.2	事件处理函数	77
6.4.3	事件绑定	77
	本章小结	79
第 7 章	数据库应用	80
7.1	结构化查询语言 (SQL)	80
7.1.1	SQL 基本语句	80
7.1.2	SQL 查询语句	81
7.2	SQLite 3 数据库基础	83
7.3	Python 的 SQLite3 数据库编程	84
7.3.1	访问数据库的步骤	84
7.3.2	数据库应用实例	85

7.4 Python 访问主流数据库与存储文本数据	93
7.4.1 Python 访问主流数据库	93
7.4.2 Python 存储文本数据	95
本章小结	95
第8章 网络及多线程	96
8.1 网络编程基础	96
8.1.1 客户端/服务器	96
8.1.2 IP 协议	96
8.1.3 TCP 和 UDP 协议	97
8.1.4 端口	97
8.1.5 socket	97
8.2 TCP 编程	102
8.2.1 TCP 客户端编程	102
8.2.2 TCP 服务器端编程	104
8.3 UDP 编程	106
8.4 多线程编程	107
8.4.1 进程和线程	107
8.4.2 创建线程	108
8.4.3 线程同步	112
本章小结	114
第9章 网络爬虫	115
9.1 网络爬虫基本原理	115
9.2 requests 库	116
9.2.1 requests 库的安装	116
9.2.2 requests 库的使用方法	117
9.2.3 访问异常处理	121
9.2.4 requests 库的应用案例	122
9.3 BeautifulSoup 库	125
9.3.1 BeautifulSoup 库的安装	125
9.3.2 BeautifulSoup 库的使用方法	126
9.3.3 BeautifulSoup 库的应用案例	132
本章小结	142
第10章 图像操作与处理	143
10.1 图像处理类库 Pillow 的安装	143
10.2 图像处理类库 PIL 的基本概念	144
10.3 图像处理类库 PIL 的常用模块	146

10.3.1	Image 模块	146
10.3.2	ImageChops 模块	147
10.3.3	ImageDraw 模块	149
10.3.4	ImageEnhance 模块	152
10.3.5	ImageFilter 模块	155
10.3.6	ImageFont 模块	156
10.4	PIL 对图像的基本操作	157
10.4.1	图像格式的转换	157
10.4.2	创建缩略图	158
10.4.3	图像的复制和粘贴	159
10.4.4	几何变换	160
10.5	PIL 对图像的综合实例	160
	本章小结	163
第 11 章	科学计算与数据可视化应用	164
11.1	NumPy 库的使用	164
11.1.1	NumPy 数组的使用	164
11.1.2	NumPy 数组的算术运算	166
11.1.3	NumPy 数组的形状操作	168
11.1.4	NumPy 文件存取数组操作	169
11.1.5	NumPy 的图像数组操作	171
11.2	Matplotlib 数据可视化	174
11.2.1	使用 Matplotlib.pyplot 模块绘图	174
11.2.2	使用 matplotlib.pylab 模块绘制直方图、条形图、散点图、饼状图等	180
	本章小结	186
第 12 章	数据挖掘与机器学习	187
12.1	机器学习概念与操作流程	187
12.1.1	概念与定义	187
12.1.2	AI 操作流程	188
12.1.3	机器学习库 sklearn 的安装	188
12.2	Python 机器学习算法与应用	188
12.2.1	线性回归算法	189
12.2.2	KMeans 聚类算法	190
12.2.3	kNN 近邻算法	191
12.2.4	逻辑回归算法	192
	本章小结	194

第 13 章 Python 解析 XML	195
13.1 XML 概述	195
13.1.1 XML 简介	195
13.1.2 Python 解析 XML 文件的两种常用方法	196
13.2 Python 使用 SAX 解析 XML	197
13.2.1 使用 SAX 解析 XML 文件的方法	197
13.2.2 使用 SAX 读取 XML 文件的实例	197
13.3 Python 使用 DOM 解析 XML	198
13.3.1 使用 DOM 读取 XML 文件	198
13.3.2 使用 DOM 添加结点	200
13.3.3 使用 DOM 修改、删除结点	201
本章小结	203
参考文献	204

导学

Python 是一种跨平台的、开源的解释型高级语言。其作为动态编程语言更适合初学者。通过引用外部库，Python 可快速、准确地实现多种实用功能，尤其在人工智能和大数据方面的应用，更优于目前市面上流行的其他高级程序语言。Python 模块化的设计理念，使其具有更好的开放性和扩展性。通过学习 Python，不但能够帮助初学者掌握程序设计方法，增强逻辑思维能力，还可使学生快速、方便地开发出具有实用功能的应用案例。

了解：Python 的发展历史和主要应用领域。

掌握：Python 的相关概念以及特点，Python 及其集成开发环境的搭建、工作方式及安装方法，Python 主要的输入和输出方法。

Python 作为一种高扩展性的语言，有功能丰富的标准库，应用领域十分广泛，几乎可以在任何场合应用 Python，从网站、爬虫到机器人控制，它是目前最主流的程序设计语言。

1.1 Python 简介

1.1.1 Python 的概念

Python 是一种高层次的，结合了解释性、编译性、互动性和面向对象的程序设计语言。Python 的设计具有很强的可读性，具有比其他语言更有特色的语法结构。

(1) Python 是一种解释型语言：开发过程中没有编译这个环节，类似于 PHP 或者 Perl 语言。

(2) Python 是交互式语言：可以在命令提示符中，直接互动执行程序。

(3) Python 是面向对象语言：Python 支持面向对象的风格或代码封装在对象中的编程技术。

(4) Python 是被广泛应用的语言：Python 支持广泛的应用程序开发，从简单的文字处理到浏览器再到网络爬虫、机器学习等。

1.1.2 Python 的发展历史

Python 是 Guido van Rossum 在 1989 年圣诞节开发的一门语言，这个名字来自于他最钟爱的电视剧 *Monty Python's Flying Circus*。Python 本身也是由诸多其他语言发展而来的，包括 C、C++、Unix shell 和其他的脚本语言等。

自从 2004 年以后，Python 的使用率呈线性增长。2011 年 1 月，Python 被 TIOBE 编程语言排行榜评为 2010 年度语言。

1.1.3 Python 3

Python 的 3.0 版本常被称为 Python 3000，或简称 Py3k。相对于 Python 的早期版本，这是一次较大的升级。Python 3.0 在设计的时候没有考虑向下相容，即许多针对早期 Python 版本设计的程序都无法在 Python 3.0 上正常执行。为了兼容现有程式，Python 2.6 作为一个过渡版本，基本使用了 Python 2.x 的语法和库，同时考虑了向 Python 3.0 的迁移，允许使用部分 Python 3.0 的语法与函数。新的 Python 程序建议使用 Python 3.0 版本的语法。大多数第三方库都正在努力地相容 Python 3.0 版本，即使无法立即使用 Python 3.0，也建议编写相容 Python 3.0 版本的程式，本书中所有 Python 代码均采用 Python 3 编写。

1.1.4 Python 的特点

1. Python 的优点

Python 作为目前应用广泛的程序设计语言，具有如下优点。

- (1) 易于学习：Python 有相对较少的关键字，结构简单，语法定义明确，学习起来更加简单。
- (2) 易于阅读：Python 代码定义得更清晰。
- (3) 易于维护：Python 的成功在于它的源代码是相当容易维护的。
- (4) 广泛的标准库：Python 的最大优势之一是丰富的、跨平台的库，在 UNIX、Windows 和 Macintosh 上的兼容性很好。
- (5) 互动模式：支持互动模式，可以终端输入执行代码并获得结果，互动地测试和调试代码片段。
- (6) 可移植性：基于其开发源代码的特性，Python 已经被移植到许多平台。
- (7) 可扩展性：Python 程序中可调用 C 或者 C++ 程序。
- (8) 数据库：Python 提供所有主要的商业数据库的接口，如 MySQL、MongoDB 等。
- (9) 可嵌入性：可以将 Python 嵌入到 C 或者 C++ 程序，让程序获得“脚本化”的能力。

2. Python 的缺点

虽然 Python 具有诸多优点，但同样存在以下缺点。

- (1) 运行速度慢：和 C 程序相比非常慢，因为 Python 是解释型语言，代码在执行时会逐行翻译成 CPU 能理解的机器码，这个翻译过程非常耗时，所以很慢，而 C 程序是运行前直接编译成 CPU 能执行的机器码，所以非常快。
- (2) 代码不能加密：如果要发布 Python 程序，实际上就是发布源代码，这一点跟 C 语言不同。C 语言不用发布源代码，只需要把编译后的机器码发布出去，要从机器码反推出 C 代码是不可能的，所以，凡是编译型的语言都没有这个问题，而解释型的语言则必须把源码发布出去。

1.1.5 Python 的应用领域

Python 作为一种功能强大且通用的编程语言而广受好评，它具有非常清晰的语法特点，适用于多种操作系统，目前在国际上非常流行，正在得到越来越多的应用。Python 具有优

秀的扩展性，在诸多领域，如网络爬虫、人工智能、科学计算、Web 开发、系统运维、大数据、云计算、金融、图形界面、企业和网站等方面都具有良好的应用。

1.2 Python 的安装

Python 具有良好的兼容性，可安装在 UNIX、Windows 和 Macintosh 等主流平台上，本节以 Windows 平台为例，详细介绍 Python 的安装过程。

首先进入 Python 官网下载页面 <https://www.python.org/downloads/>，如图 1-1 所示。

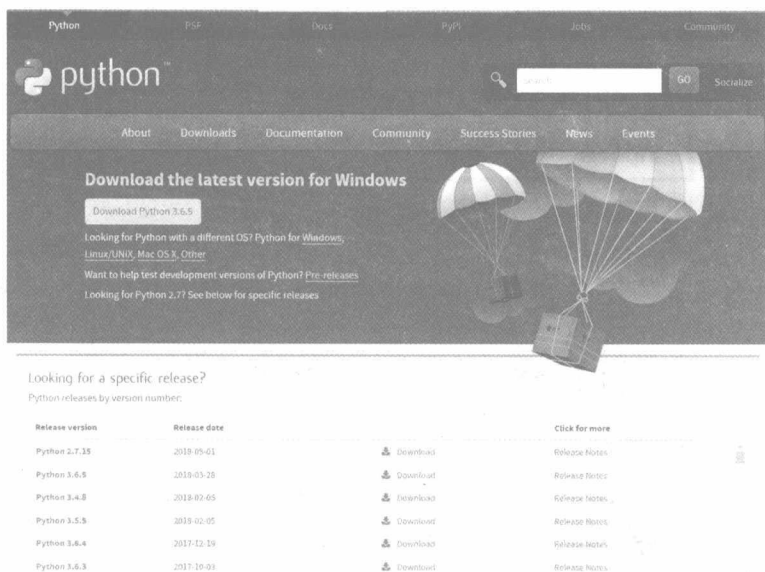


图 1-1 Python 官网下载页面

单击 Download Python 3.6.5 按钮，进入下载页面，网站会根据操作系统下载合适的版本，如图 1-2 所示。

双击安装包进行安装，在安装界面勾选 Add Python 3.6 to PATH，将 Python 添加到环境变量，如图 1-3 所示。



图 1-2 下载 Python 安装包



图 1-3 勾选 Add Python 3.6 to PATH

可选择 Install Now 进行默认安装，也可选择 Customize installation 进行个性化安装，本节以个性化安装为例进行安装，单击 Customize installation 按钮，进入个性化安装页面，可勾选需要安装的模块，如图 1-4 所示。

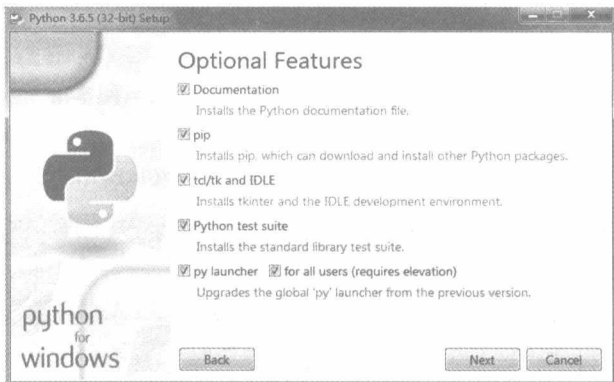


图 1-4 个性化安装界面 1

单击 Next 按钮继续安装，进入 Advanced Options 界面，可进行安装目录选择等设置，如图 1-5 所示。

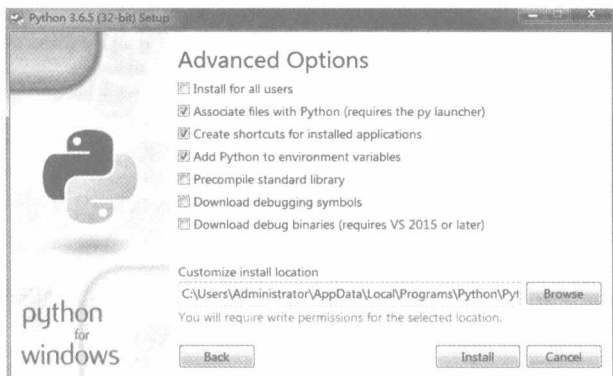


图 1-5 个性化安装界面 2

单击 Install 按钮进行安装，如图 1-6 所示。

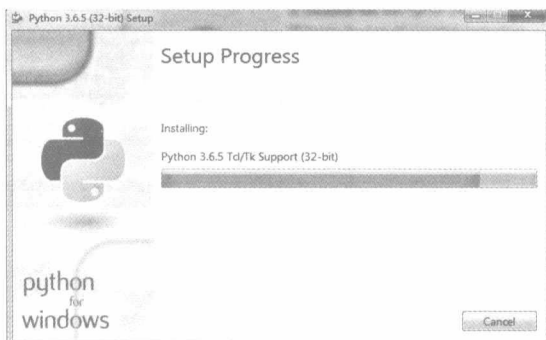


图 1-6 Python 安装界面

安装完成后会弹出安装成功界面，如图 1-7 所示。



图 1-7 Python 安装成功界面

可在 cmd 窗口测试 Python 是否安装成功。在 cmd 窗口输入 python 命令，如果显示版本等信息则说明 Python 安装成功，如图 1-8 所示。

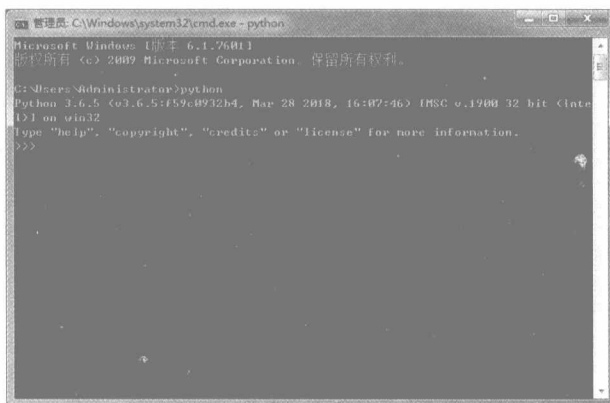


图 1-8 在 cmd 窗口测试 Python

1.3 第一个 Python 程序

对于大多数编程语言，第一个入门编程代码便是"Hello World！"，以下代码为使用 Python 输出"Hello World！"：

```
print("Hello, World!")
```

以上命令输出结果如下：

```
Hello, World!
```

这样便编写了一个 Python 程序，成功输出了"Hello World！"文本，在接下来的章节中会详细讲解 Python 的使用语法，本节作为示例，不对代码进行详细说明。

1.4 搭建 Python 集成开发环境

在成功安装 Python 后，可在 cmd 窗口运行 Python 程序。但在 cmd 窗口运行程序存在效率较低、易出错等缺点。一般在开发 Python 程序时，会选择集成开发环境(IDE, Integrated Development Environment)。PyCharm 是一种使用广泛的 Python 集成开发环境，可帮助用户在使用 Python 语言开发时提高其效率，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。此外，该集成开发环境提供了一些高级功能，以用于支持 Django 框架下的专业 Web 开发。本节以 PyCharm 为例，讲解 Python 集成开发环境的搭建。

首先进入 PyCharm 官网 <http://www.jetbrains.com/pycharm/>，如图 1-9 所示。



图 1-9 PyCharm 官网

单击 DOWNLOAD NOW 按钮，进入下载版本选择页面，可选择相应操作系统以及不同版本进行下载，如图 1-10 所示。



图 1-10 下载版本选择页面

Professional 版本具有 Python 开发的全部功能,单击 Professional 下的 DOWNLOAD 按钮,弹出安装包下载页面,如图 1-11 所示。



图 1-11 下载 PyCharm 安装包

双击 PyCharm 安装包进行安装,如图 1-12 所示。

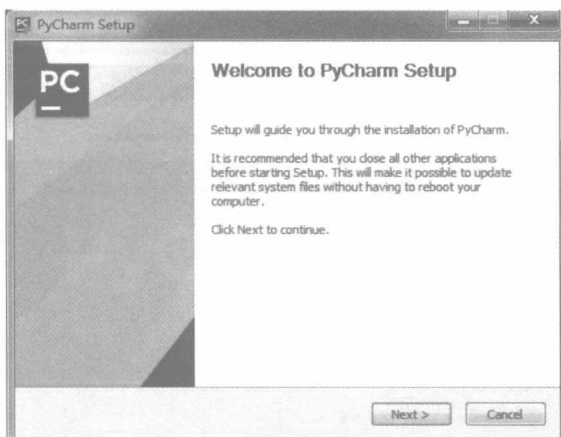


图 1-12 PyCharm 安装界面

单击 Next 按钮进行安装,选择安装目录,如图 1-13 所示。

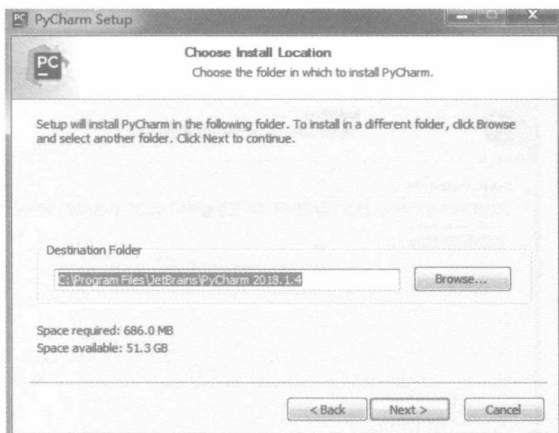


图 1-13 PyCharm 目录选择界面

单击 Next 按钮继续安装,可进行快捷方式和文件关联等配置,如图 1-14 所示。