

高等学校**计算机**教材

• 含典型案例视频分析 •

Python 实用教程

郑阿奇 / 主编



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

高等学校计算机教材

Python 实用教程

(含典型案例视频分析)

郑阿奇 主编



电子工业出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书共五个部分。第一部分（前 10 章）为基础篇，介绍 Python 概述、语言基础、分支和循环控制、序列（列表、元组、集合、字典）、数组和矩阵、字符串和正则表达式、函数和模块、面向对象编程、文件操作和异常处理等内容。第二部分（后 10 章）为应用篇，介绍二维图表、三维图像、声频、视频；Python 操作的数据库介绍 MySQL、PostgreSQL、SQL Server、Oracle、SQLite、MongoDB 等内容均完成了应用实例；Office 组件介绍 Word、Excel 和 PowerPoint 的内容，并有综合应用实例；介绍在 C++ 中如何使用 Python，以及在 Python 中如何使用 C++。第三部分为实验。第四部分为习题。第五部分为附录，提供 Python 调试及其实例。

本书将命令和编程结合，编程和常用算法结合，学习和应用结合；配套教程、习题、实验等，并各有侧重、相互配合；提供配套教学视频，分析 Python 典型实例。同时，通过网络配套提供所有实例源程序（.py）及其工程、数据库文件、教学课件，以方便教学和学生模仿。

本书既可作为大学本科和高职高专院校有关课程教材，也可作为 Python 学习、培训讲义。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有，侵权必究。

图书在版编目（CIP）数据

Python 实用教程：含典型案例视频分析 / 郑阿奇主编. —北京：电子工业出版社，2019.3

高等学校计算机教材

ISBN 978-7-121-36161-6

I. ①P… II. ①郑… 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（2019）第 051582 号

策划编辑：程超群

责任编辑：底 波

印 刷：涿州市京南印刷厂

装 订：涿州市京南印刷厂

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本：787×1092 1/16 印张：25 字数：672 千字

版 次：2019 年 3 月第 1 版

印 次：2019 年 3 月第 1 次印刷

定 价：69.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 zltts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：（010）88254577，ccq@phei.com.cn。

前言

2017 年,在 *IEEE Spectrum* 杂志发布编程语言排行榜,排在前位的分别为 Python、C、Java 和 C++,这让 Python 粉丝非常兴奋。

2017 年,教育部考试中心发布了《关于全国计算机等级考试(NCRE)体系调整的通知》(教试中心函〔2017〕205 号),决定自 2018 年 9 月起,在全国计算机等级考试(二级)中加入“Python 语言程序设计”科目。与 Python 热对应,关于 Python 的书籍也不少,其中一般专题的书多,国外翻译的书多,但非常适合作为教材的却并不多。根据这个基本情况,为初学者提供一个简单方便的学习讲义,为广大高校师生提供一个教学方便的教程,成为我们努力的目标,这当然需要接受市场的检验。

本书共五个部分。第一部分(前 10 章)为基础篇,介绍 Python 概述、语言基础、分支和循环控制、序列(列表、元组、集合、字典)、数组和矩阵、字符串和正则表达式、函数和模块、面向对象编程、文件操作和异常处理等内容。第二部分(后 10 章)为应用篇,介绍二维图表、三维图像、音频、视频等内容;Python 操作的数据库介绍 MySQL、PostgreSQL、SQL Server、Oracle、SQLite、MongoDB 等内容均完成了应用实例;Office 组件介绍 Word、Excel 和 PowerPoint 的内容,并有综合应用实例;另外,还介绍在 C++中如何使用 Python,以及在 Python 中如何使用 C++的相关内容。第三部分为实验。第四部分为习题。第五部分为附录,提供 Python 调试及其实例。

本书有如下主要特点:

(1) 命令和编程结合,编程和常用算法结合。书中每一个命令均在 Python3.x IDLE 环境下执行过,执行结果根据情况直接显示,或者在语句后给出注释。每个程序都在 PyCharm (2018 版)程序设计环境下验证通过,并且一般都包含运行结果的屏幕截图。

(2) 学习和应用结合。前 10 章是基础,并内含小应用。后 10 章是应用,为了方便学习,同步介绍了必备知识。这样,读者在学习 Python 的同时可以学到更多应用场景知识和编程方法。

(3) 本书配套教程、实验、习题等各有侧重、相互配合,可作为学习、培训讲义,特别方便作为大学本/专科有关课程教材。

(4) 配套提供教学视频,通过扫描二维码播放,介绍和分析典型 Python 实例,可更好地理解 Python 及其应用。

(5) 提供配套的网络资源,包括本书所有实例源程序(.py)及其工程,并且清楚标注出代码对应的文件名、工程名、目录名,方便读者查找。提供本书操作的数据库文件、所有章节的教学课件,方便教学和学生模仿。

本书配套资源均免费提供,需要者可通过华信教育资源网(www.hxedu.com.cn)免费下载。

本书既可作为大学本科、高职高专院校相关课程的教材和教学参考书,也可供从事 Python 应用系统开发的用户学习和参考。

本书由郑阿奇(南京师范大学)主编,参加编写的还有周何骏、孙德荣、王钢花、刘美芳、卢霞、秦洪林、刘博文、郑博琳、刘忠等,在此一并表示感谢!

由于编者水平有限,疏漏和错误在所难免,敬请广大师生、读者批评指正,意见和建议可反馈至编者电子邮箱 easybooks@163.com。

编者

本书视频目录

序 号	文 件 名	视频所在章节
1	绘制螺旋曲线实例.mp4	第 11 章
2	演示摆线形成实例.mp4	第 11 章
3	斐波那契法计算黄金分割数.mp4	第 12 章
4	绘制圆柱体.mp4	第 13 章
5	文件载入“小胡巴”.mp4	第 13 章
6	电子衍射图案.mp4	第 14 章
7	模拟穿越虫洞.mp4	第 14 章
8	地月系引力场.mp4	第 14 章
9	蝴蝶效应演示.mp4	第 14 章
10	Python 操作 MySQL.mp4	第 15 章
11	Python 操作 SQLite.mp4	第 15 章
12	Python 操作 MongoDB.mp4	第 15 章
13	Python 操作 PostgreSQL.mp4	第 15 章
14	人员信息管理系统.mp4	第 16 章
15	用 Qt 设计 Python 程序界面.mp4	第 16 章
16	Tkinter 界面呈现 Matplotlib 图表.mp4	第 16 章
17	爬虫获取天气预报.mp4	第 17 章
18	统计并演示中国高等教育普及率.mp4	第 18 章
19	长白山天池水怪研究.mp4	第 19 章
20	海洋馆潜水员表演视频剪辑.mp4	第 20 章

目 录

第一部分 基础篇

第 1 章 Python, 掀起你的盖头来	1	2.3.1 输出到屏幕	31
1.1 Python 简介	1	2.3.2 键盘输入	33
1.2 安装 Python 及其扩展库	2	2.4 综合应用实例	34
1.2.1 安装 Python	2	第 3 章 这样就能控制: 分支和循环	35
1.2.2 Python 集成开发环境——IDLE	5	3.1 选择结构	35
1.3 PyCharm 开发环境的安装和设置	5	3.1.1 if 语句的三种形式	35
1.3.1 PyCharm 的安装	6	3.1.2 if 语句的嵌套	38
1.3.2 PyCharm 程序设计环境	9	3.2 循环结构	38
1.3.3 一个简单的程序实例	11	3.2.1 循环语句	38
1.4 扩展库的安装和使用	13	3.2.2 循环控制语句	40
1.4.1 扩展库的安装	13	3.2.3 循环嵌套	41
1.4.2 扩展库的导入和使用	15	3.3 范围和迭代器	42
1.4.3 编程环境同步	16	3.3.1 范围	42
1.5 Python 说明	16	3.3.2 迭代器	43
1.5.1 程序语法规则	16	3.4 综合应用举例	43
1.5.2 Python 语言的执行	19	第 4 章 序列是什么: 列元字集	46
1.5.3 name 属性的作用	19	4.1 列表: list	46
第 2 章 Python 虽神, 语言基础并不特别	20	4.1.1 创建列表	46
2.1 数据类型	20	4.1.2 索引和切片	47
2.1.1 数值	20	4.1.3 列表的基本操作	48
2.1.2 字符串与字节串	22	4.1.4 列表方法	49
2.1.3 数据类型转换	23	4.1.5 列表推导式	52
2.1.4 类型变量	25	4.2 元组: tuple	54
2.2 运算符与表达式	26	4.2.1 元组的基本操作	54
2.2.1 算术运算符及其表达式	26	4.2.2 生成器推导式	55
2.2.2 关系运算符及其表达式	27	4.3 集合: set	56
2.2.3 位运算符及其表达式	27	4.3.1 集合的创建与访问	56
2.2.4 逻辑运算符及其表达式	28	4.3.2 集合的基本操作符	57
2.2.5 字符串基本运算及其表达式	29	4.4 字典: dict	59
2.2.6 赋值运算符	30	4.4.1 字典的基本操作符	59
2.2.7 与同性判断运算符	31	4.4.2 字典方法	60
2.3 基本输入和输出	31		

4.5 序列	62	6.4.4 综合应用实例	98
4.5.1 序列间的转换	62	第7章 代码重用和共享：函数和模块	100
4.5.2 常用内置函数	64	7.1 Python 程序结构	100
4.6 综合应用实例	67	7.2 用户编写函数	101
第5章 又是数组，又是矩阵	70	7.2.1 函数定义与调用	101
5.1 数组	70	7.2.2 函数嵌套定义	102
5.1.1 创建数组	70	7.2.3 修饰器	102
5.1.2 元素的增加、查询、修改、 删除	71	7.2.4 列表推导式	103
5.1.3 分割切片	73	7.3 参数传递	103
5.1.4 运算	73	7.3.1 定长参数	104
5.1.5 条件函数	74	7.3.2 可变长度参数	105
5.2 矩阵	75	7.3.3 序列解包	105
5.2.1 创建矩阵	75	7.4 函数嵌套与递归	106
5.2.2 常用的矩阵运算	76	7.5 变量作用域	107
5.2.3 常用的矩阵求值	77	7.5.1 局部变量	107
5.2.4 矩阵的分隔和合并	78	7.5.2 全局变量	108
5.2.5 矩阵和数组	79	7.6 lambda 表达式	108
5.3 array 模块数据库基本概念	80	7.7 成器函数设计	110
第6章 虽是字符串，还要正则表达式	83	7.8 模块	111
6.1 基本说明	83	7.9 命名空间	112
6.1.1 字符串编码	83	7.9.1 命名空间的分类	112
6.1.2 转义字符	84	7.9.2 命名空间的规则	113
6.1.3 字符串常量	85	7.10 包	113
6.2 字符串格式化	86	7.10.1 包介绍	114
6.2.1 用%符号进行格式化	86	7.10.2 包管理工具	114
6.2.2 用 format()方法格式化	87	7.11 综合应用实例	115
6.2.3 格式化的字符串常量	88	第8章 对象需要谈一谈：面向对象编程	117
6.2.4 用 template 模板格式化	89	8.1 类	117
6.3 字符串常用操作	89	8.1.1 基本概念	117
6.3.1 字符串创建和访问	89	8.1.2 类的定义与使用	118
6.3.2 字符串操作和判断	91	8.1.3 数据成员	119
6.3.3 字符串排版和切片	93	8.1.4 方法	120
6.3.4 综合应用实例	93	8.1.5 属性	122
6.4 正则表达式	94	8.1.6 动态性	123
6.4.1 正则表达式元字符	94	8.2 继承、多态	124
6.4.2 正则表达式模块 re	95	8.2.1 继承	124
6.4.3 match 对象和正则表达式子 模式	98	8.2.2 多重继承	125
		8.2.3 多态	126
		8.3 运算符重载	127

第9章 信息永久保存：文件操作	129	9.4 文件和文件夹的操作	135
9.1 文件及其操作	129	9.4.1 文件操作	135
9.1.1 文件类型	129	9.4.2 目录操作	135
9.1.2 文件的打开和关闭	130	第10章 野马不会脱缰：异常处理	137
9.2 文件操作	131	10.1 异常的产生	137
9.2.1 写入方法	131	10.2 内置异常类	137
9.2.2 读取操作方法	132	10.3 异常处理结构	138
9.2.3 定位与移动	132	10.4 抛出异常	142
9.2.4 文件设备	133	10.5 自定义异常	143
9.2.5 上下文管理语句	134	10.6 断言语句	143
9.3 序列化和反序列化	134		

第二部分 应用篇

第11章 拿起一支笔画来画去：二维图表		12.3.1 内置 ufunc 函数	165
实例	145	12.3.2 自定义 ufunc 函数：提取任意范围素数	166
11.1 单幅图表：绘制螺旋曲线实例	145	12.3.3 ufunc 函数的性能	167
11.1.1 背景知识	145	12.4 综合应用实例：斐波那契法计算黄金分割数	169
11.1.2 实现原理	146	12.4.1 背景知识	169
11.1.3 程序及分析	147	12.4.2 算法设计	170
11.2 绘制多轴图表：展示初等函数实例	148	12.4.3 编程实现	172
11.2.1 背景知识	148	第13章 进入立体世界：三维可视化实例	176
11.2.2 实现原理	149	13.1 3D 绘图入门：绘制圆柱体	176
11.2.3 程序及分析	150	13.1.1 基本概念	176
11.3 定制表轴刻度：演示摆线形成实例	152	13.1.2 绘图流程	177
11.3.1 背景知识	152	13.1.3 VTK 绘制圆柱体	179
11.3.2 实现原理	153	13.1.4 TVTK 绘制圆柱体	180
11.3.3 程序及分析	153	13.2 TVTK 绘图进阶	182
第12章 为什么这么快：数值处理及实例	156	13.2.1 流水线浏览器：改变圆柱外观	182
12.1 数组的创建	156	13.2.2 多实体场景：绘制多种几何体	185
12.1.1 NumPy 数组函数	156	13.3 使用第三方模具：文件载入“胡巴”	187
12.1.2 数组维度设定	157	13.3.1 背景知识	187
12.1.3 函数生成数组	158	13.3.2 程序实现	188
12.2 数组的存取	159	13.3.3 控制照相机	190
12.2.1 基本存取方法	159		
12.2.2 多维数组的存取	160		
12.2.3 结构数组的存取：读取修改学生成绩	163		
12.3 ufunc 函数	165		

第 14 章 精彩纷呈：可视化科学计算实例	192	16.1.3 界面的事件响应	236
14.1 SciPy 科学计算功能	192	16.2 窗体基本控件	237
14.1.1 数据拟合	192	16.2.1 Tkinter 控件概述	237
14.1.2 插值优化	193	16.2.2 标签	238
14.1.3 信号处理	194	16.2.3 复选框	238
14.2 Mayavi 可视化应用	195	16.2.4 单选按钮	239
14.2.1 绘制空间曲面：电子衍射 图案	195	16.2.5 标签组框	240
14.2.2 复杂三维图形：模拟穿越 虫洞	198	16.2.6 输入框	241
14.2.3 标量场可视化：地月系引 力场	201	16.2.7 列表框与滚动条	243
14.3 综合应用实例：蝴蝶效应演示	203	16.3 综合应用实例：人员信息管理系统	244
14.3.1 背景知识	203	16.3.1 数据库准备	244
14.3.2 实现原理	204	16.3.2 系统实现	245
14.3.3 程序演示	204	16.3.3 功能演示	248
第 15 章 流行数据库世界：操作实例	208	16.4 用 Qt 设计 Python 程序界面	250
15.1 基本原理	208	16.4.1 Qt 简介及功能展示	250
15.1.1 Python 访问数据库原理	208	16.4.2 用 Qt 设计图形界面	252
15.1.2 数据库操作的一般步骤	210	16.4.3 Qt 界面向 Python 转化	256
15.2 Python 操作 MySQL	213	16.4.4 Python 添加功能逻辑	257
15.2.1 环境安装	214	16.5 Tkinter 界面呈现 MatPlotLib 图表	260
15.2.2 程序及分析	215	第 17 章 网络爬虫，爬来爬去实例	262
15.3 Python 操作 SQLite	217	17.1 爬虫概述	262
15.3.1 SQLite 简介	217	17.1.1 权限及试验用网站	262
15.3.2 程序及分析	217	17.1.2 爬虫工作的基本流程	263
15.4 Python 操作 MongoDB	219	17.2 基于 Requests 的爬虫	266
15.4.1 环境安装	219	17.2.1 环境安装	267
15.4.2 程序及分析	221	17.2.2 程序实现	268
15.5 Python 操作 PostgreSQL	223	17.3 Python 内置 Urllib 爬虫库	269
15.5.1 环境安装	223	17.3.1 获取导航栏标题	270
15.5.2 程序及分析	225	17.3.2 搜索特定关键词	271
15.6 Python 操作其他数据库	228	17.3.3 抓取页面上的图片	273
15.6.1 Python 操作 SQL Server	228	17.4 综合应用实例：爬虫获取天气预报	275
15.6.2 Python 操作 Oracle	230	17.4.1 定位资源路径	275
第 16 章 窗口从何而来：界面设计实例	234	17.4.2 程序实现	277
16.1 界面编程入门	234	第 18 章 到 Office 晃一晃：操作 Excel/Word/ PowerPoint 实例	279
16.1.1 编程的一般步骤	234	18.1 Python 操作 Excel	279
16.1.2 第一个 Tkinter 程序	235	18.1.1 基本操作	279
		18.1.2 单元格操作与计算：统计 高考录取人数与录取率	280

18.2	Python 操作 Word	282	19.3.1	图像截取：人物肖像轮廓 提取	308
18.2.1	环境安装	282	19.3.2	画面抓拍：艺术体操表演 抓拍	310
18.2.2	基本读/写	283	19.4	综合应用实例：长白山天池水怪 研究	313
18.2.3	载入文档表格：读取历年 高考统计信息	284	19.4.1	背景知识	313
18.2.4	输出文档表格：2013—2017 年高考信息统计表	286	19.4.2	处理水怪的影像	315
18.3	Python 操作 PowerPoint	287	19.4.3	观察、研究及结论	316
18.3.1	环境安装	287	第 20 章	其他也精彩：音频、动画、视频与 C++实例	318
18.3.2	第一张幻灯片	289	20.1	音频处理：流行歌曲串烧	318
18.3.3	读取幻灯片中的表格：2013— 2017 年高考人数和录取率	290	20.1.1	歌曲介绍	318
18.3.4	绘制柱状图表：画出 2008— 2012 年高考报名人数柱 状图	292	20.1.2	Python 音频处理模块	319
18.4	综合应用实例：统计并演示全国 高等教育普及率	293	20.1.3	用 Python 实现歌曲串烧	319
18.4.1	原始 Word 数据准备	293	20.1.4	合成的新曲	322
18.4.2	数据转存 Excel 计算	294	20.2	动画制作：阿基米德螺线的生成 演示	323
18.4.3	输出至 PowerPoint 演示	295	20.2.1	动画相关组件	323
第 19 章	图像可以这样变化：图像处理实例	298	20.2.2	程序实现	323
19.1	Python 图片基本处理	298	20.2.3	演示效果	324
19.1.1	三种处理方式	298	20.3	视频处理：海洋馆潜水员表演 视频剪辑	324
19.1.2	模式转换：彩色照片画面 作旧	300	20.3.1	Python 视频模块	325
19.1.3	增强与滤波：海底摄影照片 美化	302	20.3.2	本例视频处理需求	325
19.2	多图合成技术	304	20.3.3	格式转换	326
19.2.1	图像合成的两种方式	304	20.3.4	冗余帧采样	328
19.2.2	插值合成：杂技演员动作 合成	305	20.3.5	必要帧处理	330
19.2.3	通道合成：人鱼美照融入 背景	306	20.3.6	特殊处理	331
19.3	图像截取与抓拍	308	20.3.7	制作片头和片尾	333
			20.3.8	合成作品	335
			20.4	Python 与 C++ 互操作	336
			20.4.1	C++调用 Python 模块功能	337
			20.4.2	Python 使用 C++的 DLL 库	341

第三部分 实 验

实验 1	345
------------	-----

实验 2	346
------------	-----

实验 3	346
实验 4	347
实验 5	348
实验 6	349
实验 7	350
实验 8	351
实验 9	352
实验 10	353
实验 11	353

实验 12	354
实验 13	355
实验 14	357
实验 15	358
实验 16	359
实验 17	360
实验 18	361
实验 19	362
实验 20	362

第四部分 习 题

习题 1	364
习题 2	365
习题 3	366
习题 4	367
习题 5	368
习题 6	369
习题 7	370
习题 8	371
习题 9	372
习题 10	373

习题 11	374
习题 12	375
习题 13	376
习题 14	377
习题 15	378
习题 16	379
习题 17	380
习题 18	381
习题 19	382
习题 20	384

第五部分 附 录

附录 A 磨刀不误砍材工——Pycharm 环境调试	
Python 程序	385
A.1 Python 程序语法错误标识	385

A.2 Python 程序调试基本方法	387
A.3 Python 实例程序运行调试	387

第一部分 基础篇

第 1 章 Python，掀起你的盖头来

1.1 Python 简介

Python 是一种开放的面向对象的解释型计算机程序设计语言，它的创始人为 Guido van Rossum。在 IEEE 发布的 2017 年编程语言排行榜中，Python 高居首位。

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，并且每 6 个月左右更新一次小版本号。Python 2.x 和 Python 3.x 之间很多用法是不兼容的，除了基本输入、输出方式有所不同外，很多内置函数和标准库对象的用法也有非常大的区别。Python 3.x 增加了很多新标准库，合并、拆分和删除了一些 Python 2.x 的标准库。适用于 Python 2.x 和 Python 3.x 的扩展库之间差别更大，所以由 Python 2.x 升级到 Python 3.x 比较困难。Python 2.7 将于 2020 年 1 月 1 日终止支持。用户如果在此之后继续得到与 Python 2.7 有关的支持，则需要付费。

1. Python 的特点

Python 能成为被广泛使用的高级程序设计语言，有如下特点：

- (1) 简单易学，免费开源。它的关键字比较少，语法有明确定义，代码清晰，源码免费开放。
- (2) 扩展简单，功能强大。它除了标准库外，还可加入第三方库，扩展能力强大。为了方便使用 Python，Anaconda、Python(x,y)、zwPython 等安装包集成了大量常用的 Python 扩展库，大幅度节约了用户配置 Python 开发环境的时间。
- (3) 方便嵌入，扬长避短。它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，能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。所以，有人喜欢把 Python 称为“胶水语言”。
- (4) 不同平台，移植应用。它能轻松地在 UNIX、Linux、Windows、Mac OS X 等平台上移植，可以运行在多种硬件平台上，并具有相同的接口。
- (5) 互动模式，解释运行。它可以从终端输入并获得结果，互动地进行测试和调试。在解释运行时不需要编译这个环节。

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，提高加载和运行速度并对源代码进行保密，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（如 Windows 平台扩展名为.exe）的可执行程序文件，从而可以脱离

Python 解释器环境和相关依赖库独立运行，并且还支持制作成.msi 安装包。

(6) 面向对象编程语言。它的代码以对象进行封装，并采用面向对象的编程技术。

2. Python 的解释器

Python 有多种类型的解释器来支持其广泛应用的工作开发，具体类型如下。

(1) CPython: 它是 Python 的官方版本，使用 C 语言实现，应用最为广泛。CPython 能将源文件（py 文件）转换成字节码文件（pyc 文件），然后在 Python 虚拟机上运行。

(2) Jython: 它是 Python 的 Java 实现。Jython 能将 Python 代码动态编译成 Java 字节码，然后在 JVM 上运行。

(3) IronPython: 它是 Python 的 C#实现。IronPython 能将 Python 代码编译成 C#字节码，然后在 CLR 上运行（与 Jython 类似）。

(4) PyPy: 它是 Python 实现的 Python，能将 Python 的字节码再编译成机器码，加快 Python 程序的运行速度。

3. Python 的应用场合

Python 应用在统计分析、移动终端开发、科学计算可视化、逆向工程与软件分析、图形图像处理、人工智能、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、计算机辅助教育、医药辅助设计、天文信息处理、化学、生物学等众多专业和领域。其中，大中型互联网企业在自动化运维、自动化测试、大数据分析、网络爬虫、Web 等方面使用 Python 较为普遍。

1.2 安装 Python 及其扩展库

Python 可应用于多平台，包括 Windows、UNIX、Linux 和 Mac OS X 等。一般的 Linux 发行版本、Mac OS X 等都自带 Python，不需要安装和配置就可直接使用，但自带的 Python 版本不是最新的。用户可以通过终端窗口输入“Python”命令查看本地是否已安装 Python 及其版本。

1.2.1 安装 Python

在 UNIX 和 Linux 平台中安装 Python 的步骤请读者自行参考有关文档。这里仅介绍在 Windows 平台安装 Python 的过程。

1. 下载 python 安装文件

从 Python 官网 (<https://www.python.org/downloads/windows/python-3.7-amd64.exe>) 中获取对应的 Python 安装文件，如图 1.1 所示。

2. 在 Windows 平台中安装 Python

(1) 要求选择 Windows 7 及以上 64 位操作系统版本。通过浏览器访问 Python 官网，在下载列表中选择 Windows 平台 64 位安装包 (Python-XYZ.msi 文件，XYZ 为版本号)。

(2) 双击下载包 (此处以 Python3.7.exe 为例)，打开 Python 安装向导，如图 1.2 所示。勾选下面 2 个选项 (其中“Add Python 3.7 to PATH”表示将 Python 安装目录加入 Windows 环境 PATH 变量路径中)。单击“Install Now”按钮 (在其下方，系统显示默认的安装目录)，系统进入 Python 的正式安装过程，如图 1.3 所示。

安装成功后的界面如图 1.4 所示。

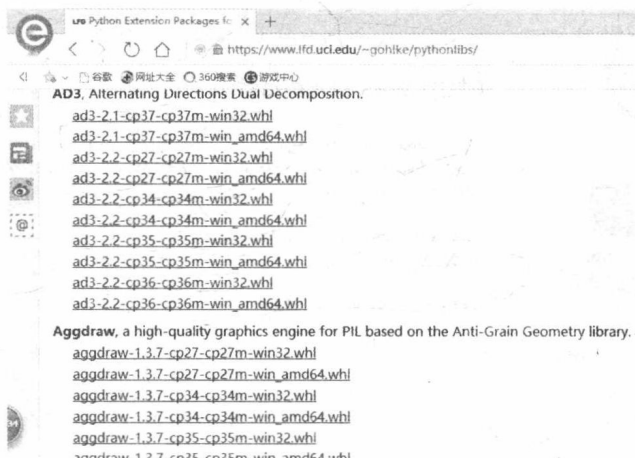


图 1.1 下载安装文件

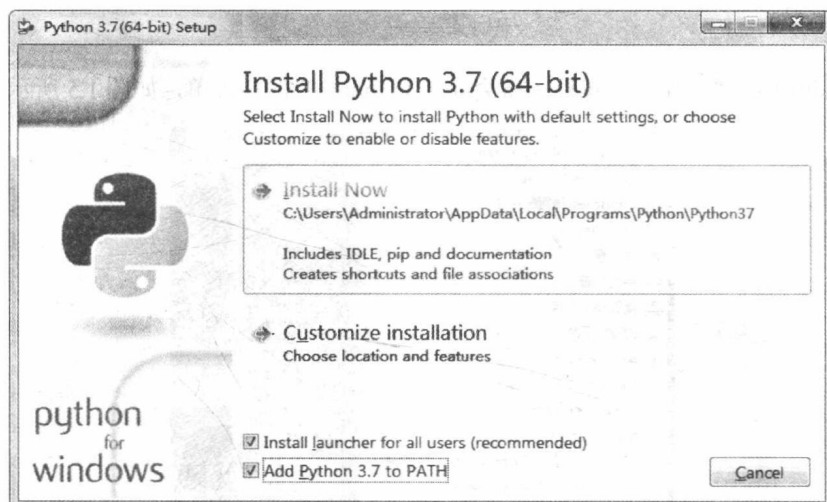


图 1.2 安装向导

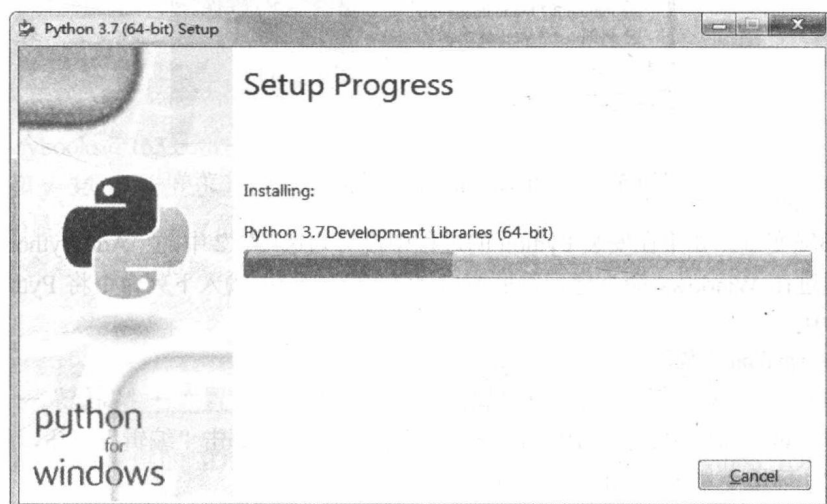


图 1.3 安装过程

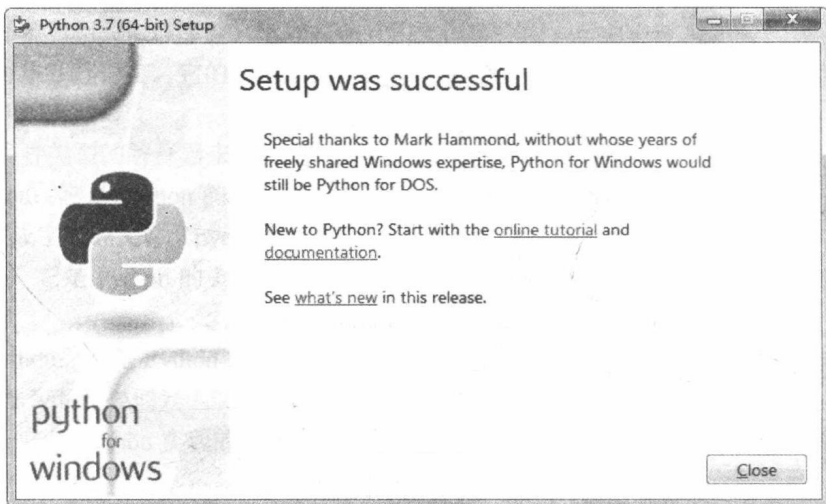


图 1.4 安装完成

此时，Windows 在“开始”菜单栏中就会显示 Python 3.7 的主菜单，如图 1.5 所示。

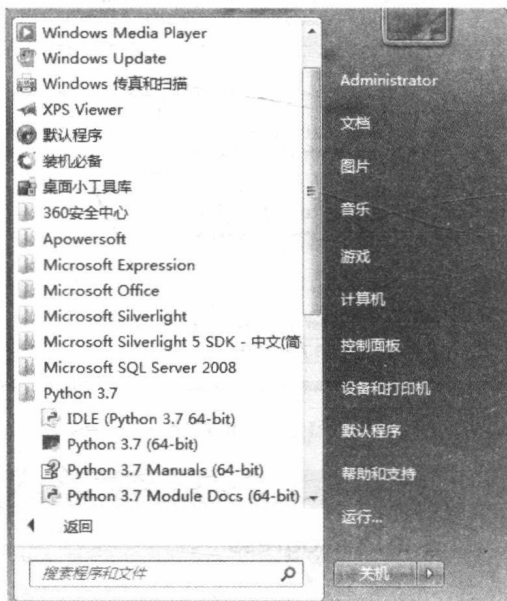


图 1.5 在“开始”菜单栏中的 Python 3.7 主菜单

(3) 设置环境变量。如果在安装 Python 的过程中没有勾选图 1.2 中的“Add Python 3.7 to PATH”选项，则需要通过在 Windows 命令提示符框窗口（运行 cmd）中输入下列命令将 Python 目录添加到 PATH 环境变量中：

```
path %path%; <python 安装目录>
```

或者，用鼠标右键单击“计算机”→“属性”→“高级系统设置”→“高级”→“环境变量”，在打开的“环境变量”窗口中选择“PATH”系统环境变量，然后单击“编辑”按钮，将 Python 安装目录添加到 PATH 环境变量中，如图 1.6 所示。

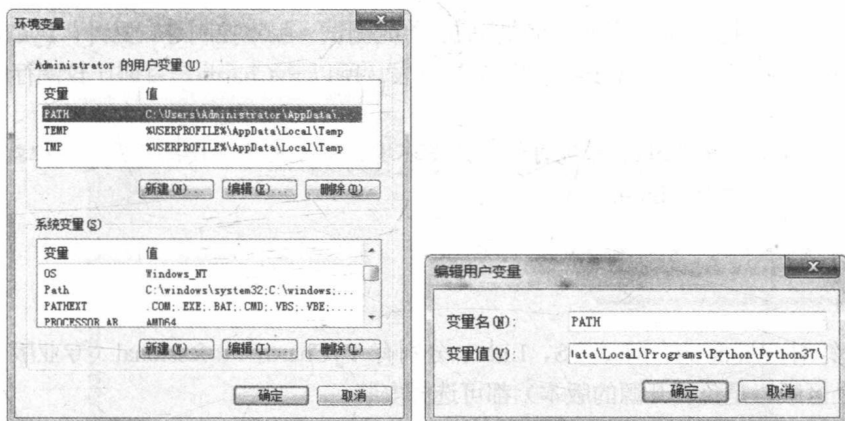


图 1.6 将 Python 安装目录添加到 PATH 环境变量中

1.2.2 Python 集成开发环境——IDLE

对于 Windows 7 操作系统，Python 3.7 安装完成后，在“开始”菜单中加入了“Python 3.7”菜单组的 6 个菜单项，其中包括“IDLE (Python 3.7 32-bit)”“IDLE (Python 3.7 64-bit)”“Python 3.7 (32-bit)”和“Python 3.7 (64-bit)”等。单击“Python 3.7 (64-bit)”或“IDLE (Python 3.7 64-bit)”项，进入 Python 3.7 (64-bit) 的 IDLE 环境窗口，系统提示符为“>>>”。在 Python IDLE 窗口中可以输入 Python 的命令或者语句，如图 1.7 所示。

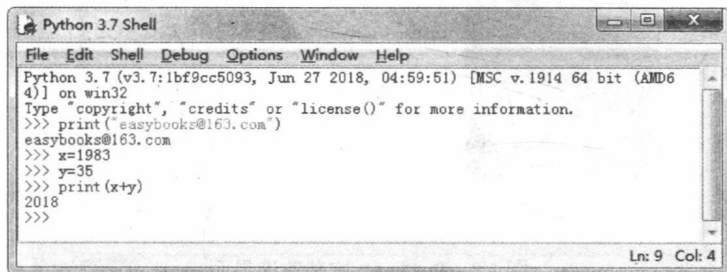


图 1.7 Python IDLE 窗口

说明：

- `print("easybooks@163.com")` 显示字符串 “easybooks@163.com”。
- `x=1983` 和 `y=35` 分别给变量 `x` 和 `y` 赋值。
- `print(x+y)` 显示表达式 `x+y` 的值。

Python 对于命令（语句的格式）特别严格，“>>>”提示符后面的空格只有一个，多了就会显示错误。

1.3 PyCharm 开发环境的安装和设置

除了 Python 官网提供的 IDLE 开发环境，还有 PyCharm、Wing Python IDE、PythonWin、Eclipse+PyDev、Eric Python IDE。

PyCharm 是由 JetBrains 打造的一款 Python IDE。它具备一般 Python IDE 的功能，如调试、语法高

亮、项目管理、代码跳转、智能提示、自动完成、单元测试、版本控制等。另外，PyCharm 还提供一些出色的功能用于 Django（一个 Web 应用框架）开发。所以 PyCharm 是目前比较流行的 Python 程序开发环境。

注意：前面的 IDLE 是 Python 系统自带的开发环境，一般用于直接执行命令，但就程序开发方便程度而言，大家比较推崇 PyCharm。

1.3.1 PyCharm 的安装

PyCharm 针对 Windows、Mac OS、Linux 分别有 PyCharm Professional（专业版）与 PyCharm Community（社区版，是免费开源的版本）都可选择安装。

以在 Windows 7 中安装 PyCharm Community 为例，简单说明 PyCharm 的安装过程。

（1）从 JetBrains 官网（<http://www.jetbrains.com/pycharm/pycharm-community-2018.1.4.exe>）中下载 PyCharm Community（社区版）。

（2）双击下载“pycharm-community-2018.1.4.exe”文件，开始安装 PyCharm Community Edition（以下简称 PyCharm），系统显示如图 1.8 所示的欢迎界面，单击“Next”按钮。

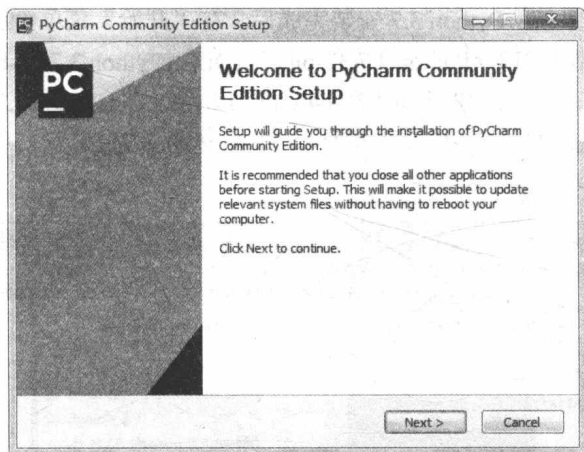


图 1.8 PyCharm 安装欢迎界面

（3）系统进入安装路径选择界面，如图 1.9 所示。

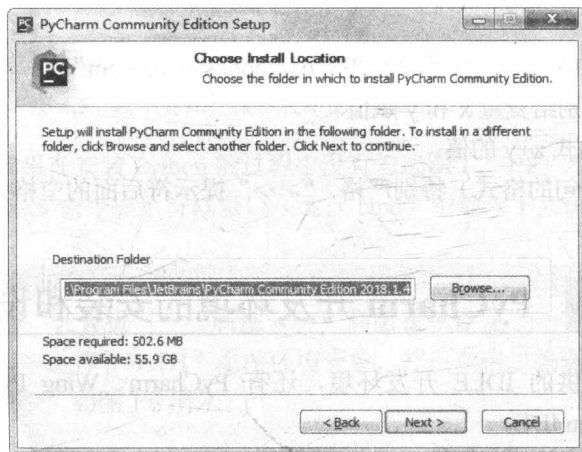


图 1.9 选择 PyCharm 安装路径