

计算机系列教材

Python 3.x 程序设计基础

周元哲 编著

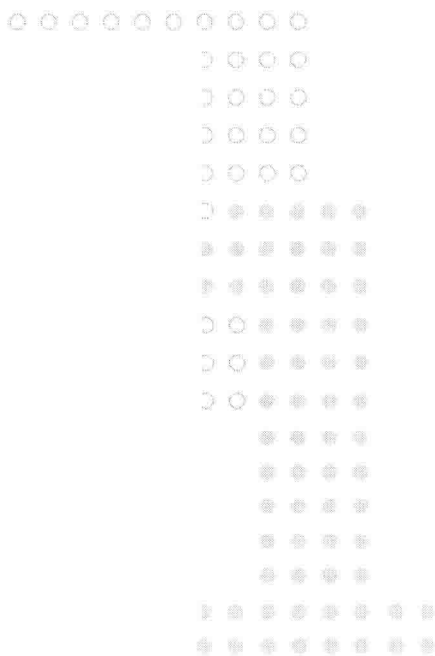
清华大学出版社



计算机系列教材

周元哲 编著

Python 3.x 程序设计基础



清华大学出版社

北京

内 容 简 介

本书以全国计算机等级考试二级 Python 语言程序设计考试大纲为依据,系统地介绍 Python 3. x 程序设计基础知识。全书共 14 章,内容包括 Python 语言概述、基本数据类型、组合数据类型、顺序与选择结构、循环结构、函数与模块、文件与数据组织、面向对象程序设计、使用 tkinter 的 GUI 设计、图形绘制、爬虫与正则表达式、SQLite 数据库、异常处理和 Python 计算生态。附录给出了全国计算机等级考试二级 Python 语言程序设计考试大纲(2018 年版)和上海市计算机等级考试二级 Python 大纲(2016 年版),以及 Python 的内置数据类型、函数和集成开发工具 IDLE。

本书适合作为高等院校相关专业 Python 程序设计课程的教材或教学参考书,也可作为全国计算机等级考试、全国计算机技术与软件专业技术资格(水平)考试的培训教材,还可供计算机应用开发领域的各类技术人员参考。

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 3. x 程序设计基础/周元哲编著. —北京:清华大学出版社,2019
(计算机系列教材)
ISBN 978-7-302-52657-5

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

中国版本图书馆 CIP 数据核字(2019)第 047101 号

责任编辑:张 民 战晓雷

封面设计:常雪影

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

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

印 刷 者:北京鑫丰华彩印有限公司

装 订 者:三河市深源装订厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:16.5 字 数:380 千字

版 次:2019 年 5 月第 1 版 印 次:2019 年 5 月第 1 次印刷

定 价:45.00 元

产品编号:081992-01

前 言

Python 已经成为继 Java、C++ 之后的第三大编程语言,作为一种面向对象的解释型计算机程序设计语言,它具有简单易学、免费开源、有丰富和强大的库等特点。本书以全国计算机等级考试二级 Python 语言程序设计考试大纲为依据,系统地介绍 Python 程序设计基础知识。全书共 14 章,内容包括 Python 语言概述、基本数据类型、组合数据类型、顺序与选择结构、循环结构、函数与模块、文件与数据组织、面向对象程序设计、使用 tkinter 的 GUI 设计、图形绘制、爬虫与正则表达式、SQLite 数据库、异常处理和 Python 计算生态。附录给出了全国计算机等级考试二级 Python 语言程序设计考试大纲(2018 年版)和上海市计算机等级考试二级 Python 大纲(2016 年版),以及 Python 的内置数据类型、函数和集成开发工具 IDLE。

学习 Python,最重要的是学习其编程思想,Python 2 和 Python 3 只存在少量的语法差异,它们的编程思想基本相同。本书的作者多年从事计算机编程语言的教学,编写了 C 语言、VB 6.0、VB.NET、Python 和程序基本算法等相关教材,一直致力于培养学生掌握编程思想及方法,以提高学生的编程应用开发能力。学习编程语言必须进行实践。希望读者注重读代码和写代码的异同点,注重编写代码能力的提高。本书的所有程序都在 Anaconda 中进行了调试和运行。

ACM-ICPC 亚洲区第一训练委员会主任吴永辉、清华大学出版社张民、西安邮电大学刘有耀、李晓戈、孟伟君、张庆生阅读了本书的部分手稿,提出了很多宝贵的意见。作者在写作本书过程中参阅了大量中英文的专著、教材、论文、报告及网络资料,在此一并向有关的作者表示衷心的感谢。

本书内容精练,结构合理,文字简洁,实训题目经典实用、综合性强,明确定位于面向初、中级读者,由零基础起步,侧重提高,特别适合作为高等院校相关专业 Python 程序设计课程的教材或教学参考书,也可以作为全国计算机等级考试、全国计算机技术与软件专业技术资格(水平)考试的培训教材,还可以供从事计算机应用开发的各类技术人员参考。

由于作者水平有限,时间紧迫,本书难免有疏漏之处,恳请广大读者批评指正。

作 者

2019 年 1 月

目 录

第 1 章 Python 语言概述	1
1.1 Python 概述	1
1.1.1 Python 的发展历程	1
1.1.2 Python 的特点	1
1.1.3 Python 的应用场合	2
1.2 Python 的安装	2
1.2.1 在 Linux 下安装 Python	2
1.2.2 在 Windows 下安装 Python	3
1.3 Python 的开发环境	5
1.3.1 IDLE	5
1.3.2 PyCharm	6
1.3.3 Anaconda	8
1.4 学习建议	14
1.5 习题	15
第 2 章 基本数据类型	16
2.1 数据类型	16
2.1.1 数字类型	16
2.1.2 字符串	18
2.2 变量	18
2.2.1 标识符	18
2.2.2 变量定义和赋值	19
2.3 运算符	20
2.3.1 算术运算符	20
2.3.2 关系运算符	21
2.3.3 赋值运算符	22
2.3.4 逻辑运算符	23
2.3.5 位运算符	23
2.3.6 成员运算符	25
2.3.7 身份运算符	25

2.4	表达式	26
2.4.1	表达式组成规则	26
2.4.2	表达式计算	26
2.4.3	注意事项	27
2.5	数据类型的转换	27
2.5.1	隐式类型转换	27
2.5.2	显式类型转换	28
2.6	Python 的关键字和转义符	29
2.6.1	关键字	29
2.6.2	转义符	29
2.7	内置函数	30
2.7.1	数学函数	30
2.7.2	随机数函数	31
2.7.3	时间函数	32
2.8	习题	33
第3章	组合数据类型	35
3.1	列表	35
3.1.1	定义	35
3.1.2	列表的成员运算符、索引和切片	38
3.1.3	操作方法	39
3.1.4	操作函数	40
3.2	元组	41
3.2.1	定义	41
3.2.2	操作方法	42
3.3	字符串	43
3.3.1	操作函数	43
3.3.2	字符串举例	45
3.4	字典	46
3.4.1	定义	46
3.4.2	字典操作	47
3.5	集合	50
3.5.1	定义	50
3.5.2	集合操作	50
3.5.3	集合运算	51
3.6	数据类型转换	52
3.7	习题	52

第 4 章 顺序与选择结构	55
4.1 程序设计流程	55
4.1.1 算法	55
4.1.2 程序流程图	57
4.1.3 3 种控制结构	58
4.2 顺序结构	58
4.2.1 输入、处理和输出	59
4.2.2 输入输出函数	59
4.3 顺序结构程序设计举例	62
4.4 选择结构	63
4.4.1 单分支结构	63
4.4.2 双分支结构	64
4.4.3 多分支结构	65
4.4.4 分支嵌套	67
4.5 选择结构程序设计举例	68
4.6 程序书写格式	69
4.6.1 缩进	69
4.6.2 多行语句	70
4.6.3 空行	70
4.6.4 注释	71
4.7 语句构造注意事项	71
4.8 习题	72
第 5 章 循环结构	73
5.1 循环概述	73
5.1.1 循环结构	73
5.1.2 循环分类	74
5.2 while 语句	74
5.2.1 基本形式	74
5.2.2 else 语句	76
5.2.3 死循环	76
5.3 for 语句	77
5.3.1 遍历循环	77
5.3.2 内置函数 range()	77
5.3.3 循环嵌套实现	78
5.4 转移语句	80
5.4.1 break 语句	81
5.4.2 continue 语句	82

5.4.3	pass 语句	83
5.5	迭代器	84
5.5.1	iter()方法	84
5.5.2	next()方法	84
5.6	循环语句举例	84
5.7	语句构造注意事项	88
5.8	习题	89
第 6 章	函数与模块	91
6.1	函数概述	91
6.1.1	函数引例	91
6.1.2	函数分类	92
6.2	函数的定义与使用	92
6.2.1	函数的定义	92
6.2.2	函数的使用	93
6.2.3	函数的返回值	94
6.3	参数传递	95
6.3.1	实参与形参	95
6.3.2	传对象引用	95
6.4	参数分类	96
6.4.1	必备参数	96
6.4.2	默认参数	97
6.4.3	关键参数	97
6.4.4	可变长参数	98
6.5	两类特殊函数	98
6.5.1	匿名函数	98
6.5.2	递归函数	99
6.6	变量作用域	103
6.6.1	局部变量	103
6.6.2	全局变量	104
6.7	模块	104
6.7.1	命名空间	104
6.7.2	模块定义与导入	105
6.8	第三方包管理工具	105
6.8.1	pip	105
6.8.2	安装 wheel 文件	106
6.8.3	将 py 文件打包成 exe 文件	107
6.9	习题	108

第 7 章 文件与数据组织	109
7.1 文件	109
7.1.1 字符编码	109
7.1.2 文本文件和二进制文件	110
7.2 文件操作	110
7.2.1 打开和关闭文件	110
7.2.2 读写文件	112
7.2.3 文件相关函数	114
7.3 文件操作举例	116
7.4 数据组织	117
7.4.1 维度	117
7.4.2 CSV 格式	118
7.5 习题	119
第 8 章 面向对象程序设计	120
8.1 面向对象概述	120
8.1.1 类与对象	120
8.1.2 三大特性	121
8.2 类属性与实例属性	122
8.2.1 类属性	122
8.2.2 实例属性	123
8.3 方法	124
8.3.1 对象方法	124
8.3.2 类方法	125
8.3.3 静态方法	126
8.4 构造函数与析构函数	126
8.4.1 构造函数	126
8.4.2 析构函数	127
8.5 继承性	127
8.6 多态性	129
8.7 习题	131
第 9 章 tkinter 的 GUI 设计	132
9.1 概述	132
9.1.1 界面设计原则	132
9.1.2 Python 的 GUI 工具	132
9.2 tkinter 概述	133
9.3 常用控件	134

9.3.1	标签	134
9.3.2	文本框	135
9.3.3	输入框	136
9.3.4	单选按钮	137
9.3.5	复选框	137
9.3.6	按钮	138
9.3.7	列表框	139
9.3.8	滚动条	139
9.3.9	对话框和消息框	140
9.4	布局	141
9.4.1	pack()方法	141
9.4.2	grid()方法	143
9.4.3	place()方法	144
9.4.4	Frame()方法	145
9.5	事件响应	146
9.6	习题	147
第 10 章	图形绘制	148
10.1	绘图简介	148
10.2	turtle	148
10.2.1	turtle 简介	148
10.2.2	绘图步骤	150
10.2.3	绘图实例	150
10.3	Canvas	154
10.3.1	Canvas 简介	154
10.3.2	绘图步骤	155
10.3.3	绘制基本图形	155
10.3.4	绘图实例	159
10.4	习题	161
第 11 章	爬虫与正则表达式	162
11.1	网络爬虫	162
11.1.1	概述	162
11.1.2	爬虫流程	162
11.2	正则表达式	162
11.2.1	基本语法	163
11.2.2	re 模块	164
11.3	Python 爬虫库	167

11.3.1	urllib 库	167
11.3.2	requests 库	168
11.3.3	BeautifulSoup 库	169
11.3.4	jieba 库	173
11.4	网络爬虫举例	177
11.4.1	需求	177
11.4.2	实现思路	177
11.4.3	实现步骤	178
11.5	习题	183
第 12 章	SQLite 数据库	184
12.1	关系型数据库	184
12.2	SQLite 数据库简介	185
12.3	sqlite3 模块操作数据库的步骤	186
12.4	SQLite 命令	187
12.5	SQLite 数据库举例	189
12.6	习题	191
第 13 章	异常处理	192
13.1	错误类型	192
13.1.1	语法错误	192
13.1.2	运行时错误	192
13.1.3	逻辑错误	193
13.2	捕获和处理异常	193
13.2.1	try...except...else 语句	193
13.2.2	try...except...finally 语句	195
13.2.3	raise 语句	196
13.2.4	自定义异常类	196
13.3	习题	198
第 14 章	Python 计算生态	199
14.1	数据分析	199
14.1.1	NumPy	199
14.1.2	SciPy	203
14.1.3	Pandas	206
14.2	数据可视化	209
14.2.1	Matplotlib 简介	209
14.2.2	绘制图形	210

14.3	Web 开发	214
14.3.1	Web 开发技术发展历程	214
14.3.2	Django 框架	215
14.4	游戏开发	217
14.4.1	Pygame 简介	217
14.4.2	Pygame 的模块	219
14.5	习题	224
附录 A 全国计算机等级考试二级 Python 语言程序设计考试大纲(2018 年版)		225
A.1	基本要求	225
A.2	考试内容	225
A.3	考试方式	226
附录 B 上海市计算机等级考试二级 Python 大纲(2016 年版)		227
B.1	考试性质	227
B.2	考试目标	227
B.3	考试细则	227
B.4	试卷结构	228
B.5	考试内容和要求	228
附录 C Python 的内置数据类型		231
附录 D Python 的内置函数		232
D.1	数学函数	232
D.2	转换函数	232
D.3	随机数函数	233
D.4	时间函数	234
D.5	列表函数	236
D.6	元组函数	236
D.7	字符串函数	236
D.8	字典函数	238
D.9	集合函数	238
附录 E Python 内置的集成开发工具 IDLE		239
E.1	IDLE 简介	239
E.2	IDLE 的两种运行方式	239
E.2.1	命令行运行方式	239
E.2.2	图形用户界面运行方式	240

E.3	IDLE 的调试方法	240
附录 F	Python 程序调试器 pdb	244
F.1	pdb 简介	244
F.2	pdb 的调用方式	244
F.2.1	在命令行调用 pdb	244
F.2.2	在 Python 交互环境中调用 pdb	244
F.2.3	pdb 模块的 set_trace 方法	245
F.3	调试命令	246
附录 G	PyCharm 编辑器	247
G.1	PyCharm 简介	247
G.2	PyCharm 调试步骤	247
	参考文献	249

第 1 章 Python 语言概述

Python 是一种解释型、面向对象、动态数据类型的高级程序设计语言,现已成为继 Java、C++ 之后的第三大编程语言。本章介绍 Python 的发展历程、特点以及应用场合,Python 在 Linux 和 Windows 下的安装方式、Python 的 3 种开发环境,给出了关于 Python 的学习建议。

1.1 Python 概述

1.1.1 Python 的发展历程

当前,全世界有 600 多种计算机编程语言,但流行的编程语言只有 20 余种。其中,C 语言适合开发涉及硬件性能的程序,Java 语言适合编写网络应用程序,BASIC 语言适合初学者,JavaScript 语言适合网页编程等。

Python 是 Guido van Rossum 在 1989 年圣诞节期间开发的,第一个公开发行版于 1991 年推出。Python 借鉴了诸多语言(如 ABC、Modula-3、C、C++、ALGOL-68、SmallTalk、UNIX Shell 和脚本语言等)的特点。Python 2.0 于 2000 年 10 月 16 日发布,实现了垃圾回收,并支持 Unicode。Python 3.0 被称为 Python 3000,或简称 Py3k,发布于 2008 年 12 月 3 日,相对于 Python 的早期版本,作了较大的升级。但 Python 3.0 未考虑向下相容,导致早期 Python 版本设计的程序无法在 Python 3.0 上正常执行。

2018 年 3 月,Python 核心团队宣布在 2020 年停止支持 Python 2,只支持 Python 3。

1.1.2 Python 的特点

Python 具有如下显著特点:

(1) 简单易学。Python 是一种代表简单主义思想的语言,具有极其简单的说明,易于快速上手学习。

(2) 免费开源。Python 是 FLOSS(Free/Libre and Open Source Software,自由/开放源码软件)之一,使用者可以自由地阅读源代码,对它做改动,把它的一部分用于新的自由软件中,以及发布这个软件的副本。

(3) 丰富的数据类型。Python 具有序列、列表、元组和字典等数据结构,便于实现各种算法。

(4) 解释型语言。计算机高级编程语言必须将源程序通过翻译程序翻译成目标程序,计算机才能识别和执行。翻译通常有两种方式:一种是编译执行,另一种是解释执行。C、C++等采用编译执行方式。编译执行是指源程序代码先由编译器编译成二进制的可执行指令。这种方式通常执行效率高。Python和Java采用解释执行方式。解释执行是指源代码程序被解释器转换成称为字节码的中间形式,由虚拟机负责在不同的计算机上运行。这种方式便于移植。

(5) 功能强大。Python在图形处理、数据分析、机器学习、科学计算、Web开发、爬虫、人工智能等领域都有所应用。

1.1.3 Python 的应用场合

Python功能强大,应用广泛,常用的应用场合有如下几种。

(1) GUI软件开发。Python具有wxPython、PyQT等工具,可以快速开发图形用户界面。

(2) 网络应用开发。Python提供了标准Internet模块,可以广泛应用到各种网络任务中。webpy、Django、flask等网络框架能够快速构建功能完善和高质量的网站。

(3) 多媒体应用。Python的PyOpenGL模块封装了OpenGL应用程序编程接口,能进行二维和三维图像处理。Pygame模块专供电子游戏设计使用。

(4) 科学计算。随着NumPy、SciPy、Matplotlib等众多程序库的开发,Python越来越适用于科学计算以及绘制高质量的2D和3D图像。相对于科学计算领域最流行的商业软件MATLAB而言,Python是一门通用的程序设计语言,其应用范围更广泛,有更多的第三方库的支持。

(5) 数据库开发。Python支持所有主流数据库,如Oracle、Sybase、MySQL、MongoDB等。

(6) 系统编程。Python对操作系统服务设置了内置接口,使其适合编写可移植的维护操作系统的管理工具和部件。Python程序可以搜索文件和目录树,可以运行其他程序,用进程或线程进行并行处理等。

1.2 Python 的安装

1.2.1 在 Linux 下安装 Python

Ubuntu(乌班图)基于Linux的免费开源桌面PC操作系统,目标在于为一般用户提供一个最新的、相当稳定的、主要由自由软件构建而成的操作系统。Ubuntu具有庞大的社区力量,用户可以方便地从社区获得帮助。

Ubuntu内置Python 2,在终端下输入Python,如图1.1所示。

在Ubuntu下安装Python 3的具体步骤如下。

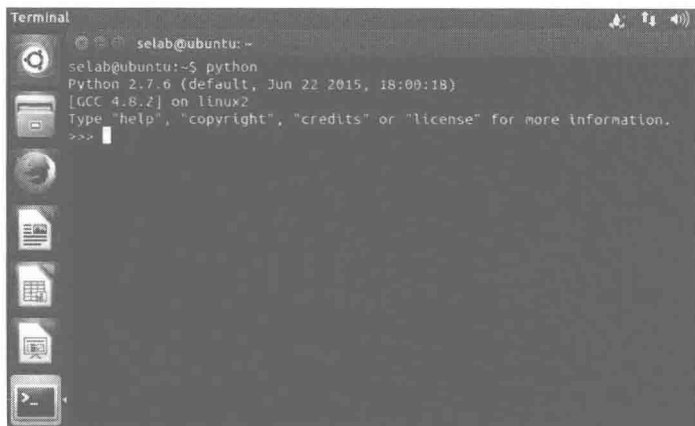


图 1.1 Ubuntu 内置 Python 2

步骤 1: 下载。

```
wget https://www.python.org/ftp/python/3.6.0/Python-3.6.0a1.tar.xz
```

步骤 2: 解压。

```
tar xvf Python-3.6.0a1.tar.xz
```

步骤 3: 创建安装文件的路径。

```
mkdir /usr/local/python3
```

步骤 4: 编译安装。

```
./configure --prefix=/usr/local/python3
```

```
make
```

```
make install
```

步骤 5: 测试。输入 Python 3 进行测试,按 Ctrl+D 键退出。

1.2.2 在 Windows 下安装 Python

在 Windows 下安装 Python 的步骤如下。

步骤 1: 下载 Python 3.6.0 安装包并进行安装。在浏览器中输入 <http://www.python.org>, 在安装文件列表中找到适合的版本下载,如图 1.2 所示。

步骤 2: 在 Windows 环境变量中添加 Python,将 Python 的安装目录添加到 Windows 下的环境变量 PATH 中,如图 1.3 所示。

步骤 3: 测试 Python 安装是否成功。

在 Windows 下使用 cmd 打开命令行,输入 python 并按 Enter 键,如果出现图 1.4 所示的信息就表示安装成功。

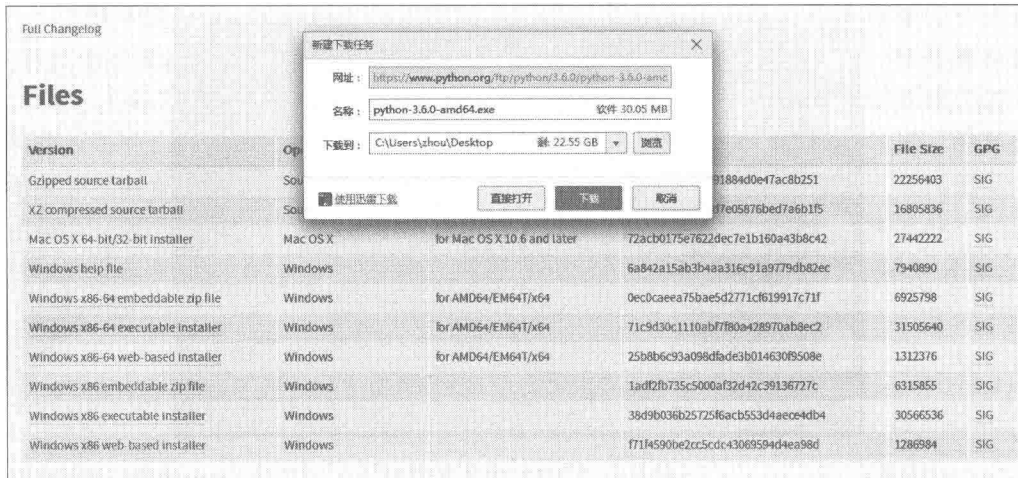


图 1.2 下载 Python 3.6.0

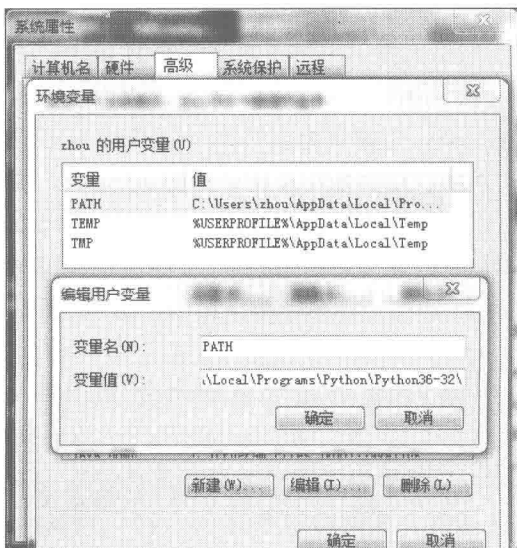


图 1.3 设置环境变量 PATH

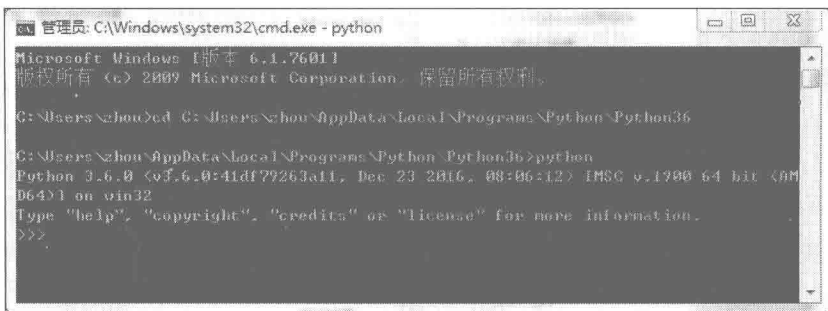


图 1.4 测试 Python 安装是否成功