



励志改变人生
编程改变命运

零基础学 Python



7小时多媒体教学视频

张志强 赵越 等编著

本书特色

- ◎ 由浅入深，循序渐进，从零开始学Python语言，一点都不难
- ◎ 编程基础、编程进阶、编程应用、项目实战"上机练习"
- ◎ 300个实例、64个练习题

超值、大容量DVD

- ◎ 本书教学视频
- ◎ 本书源代码
- ◎ 本书教学PPT

本书技术支持

- ◎ 论坛：<http://www.rzchina.net>



机械工业出版社
China Machine Press

编程
零基础学



零基础学 Python



7小时多媒体教学视频



藏志强 赵越 等编著

本书特点

1. 循序渐进，由浅入深
为了方便读者学习，本书首先让读者了解 Python 的创始人 Python 的世界，掌握 Python 语言的基本要点以及基础。
2. 技术全面，内容充实
本书在保证内容实用的前提下，内容非常全面，无论从事什么行业。
3. 对比讲解，理解深刻
有很多读者具备 Java 的开发学习，同时本书注意对 Python 中相似问题的讲解，加深读者对 Python 的理解。
4. 分析原理，讲解清晰
每种编程语言都有自己独特的特点，本书在讲解每个知识点的时候，总结实现原理和代码，帮助读者理解。
5. 代码完整，讲解详细
对于书中的每个知识点，都给出了完整的代码，读者可以参照运行结果加深对知识点的理解。



机械工业出版社
China Machine Press



图书在版编目 (CIP) 数据

零基础学 Python / 张志强等编著. —北京: 机械工业出版社, 2015.1
(零基础学编程)

ISBN 978-7-111-49211-5

I. 零… II. 张… III. 软件工具—程序设计 IV. TP311.56

中国版本图书馆 CIP 数据核字 (2015) 第 017556 号

Python 是目前最流行的动态脚本语言之一。本书由浅入深, 全面、系统地介绍了使用 Python 进行开发的各种知识和技巧。

本书内容包括 Python 环境的安装和配置、Python 的基本语法、模块和函数、内置数据结构、字符串和文件的处理、正则表达式的使用、异常的捕获和处理、面向对象的语言特性和设计、Python 的数据库编程、Tkinter GUI 库的使用、HTML 应用、XML 应用、Django 网页开发框架的使用、测试驱动开发模式应用、Python 中的进程和线程、Python 系统管理、网络编程、Python 图像处理、Python 语言的扩展和嵌入以及 Windows 下 Python 开发等。为了便于读者学习, 本书每个章节中都提供了详尽的例子, 结合实例讲解各个知识点。

本书适合 Python 爱好者、大中专院校的学生、社会培训班学生, 以及系统管理员、界面开发人员、Web 开发人员、网络编程人员等有关人员学习、使用。

零基础学Python

出版发行: 机械工业出版社 (北京市西城区百万庄大街22号 邮政编码: 100037)

责任编辑: 陈佳媛

责任校对: 殷虹

印刷: 北京瑞德印刷有限公司

版次: 2015年2月第1版第1次印刷

开本: 185mm×260mm 1/16

印张: 30

书号: ISBN 978-7-111-49211-5

定价: 79.00元 (附光盘)

ISBN 978-7-89405-698-6 (光盘)

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzjsj@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光/邹晓东

前 言

作为最流行的脚本语言之一，Python 具有内置的高级数据结构和简单有效的面向对象编程思想实现。同时，其语法简洁而清晰，类库丰富而强大，非常适合于进行快速原型开发。另外，Python 可以运行在多种系统平台下，从而使得只需要编写一次代码，就可以在多个系统平台下都保持有同等的功能。

为了能够使广大读者既能够掌握 Python 语言的基础知识，又能够将 Python 语言应用于某个特定的领域（如 Web 开发），本书将全面介绍和 Python 相关的这些内容。在学习完本书之后，相信读者能够很好地掌握 Python 语言，同时可以使用 Python 语言进行实际项目的开发。

本书特点

1. 循序渐进，由浅入深

为了方便读者学习，本书首先让读者了解 Python 的历史和特点。通过具体的例子逐渐把读者带入 Python 的世界，掌握 Python 语言的基本要点以及基础类库、常用库和工具的使用。

2. 技术全面，内容充实

本书在保证内容实用的前提下，详细介绍了 Python 语言的各个知识点。同时，本书所涉及的内容非常全面，无论从事什么行业的读者，都可以从本书中找到可应用 Python 于本行业的方

3. 对比讲解，理解深刻

有很多读者具备 Java 的开发经验，因此本书注意结合 Python 与 Java 语法的异同点进行讲解。同时本书注意对 Python 中相似的函数和方法进行对比。通过对比讲解的方式，帮助读者解决一些疑难问题，加深读者对 Python 语法要点的理解。

4. 分析原理，步骤清晰

每种编程语言都有自己独特的魅力。掌握一门技术首先需要理解原理，本书注意把握各个知识点的原理，总结实现的思路和步骤。读者可以根据具体步骤实现书中的例子，理论结合实践更利于学习。

5. 代码完整，讲解详尽

对于书中的每个知识点都有一段示例代码，并对代码的关键点进行了注释说明。每段代码的后面都有详细的分析，同时给出了代码运行后的结果。读者可以参考运行结果阅读源程序，可以加深

对程序的理解。

本书内容

第1章：如果读者还是一个新手，可通过这一章了解 Python 能做什么，Python 的特征和优势，逐渐步入 Python 的世界。

第2章：详细介绍了 Python 的语法知识，深入讲解了 Python 的编码规则、变量和常量的声明及使用、数据类型、运算符和表达式。通过本章的学习，读者能掌握 Python 编码的一些规范以及一些基本概念。

第3章：详细介绍了 Python 中的控制语句、循环语句以及一些习惯用法，结合示例讲解了 Python 结构化编程的要点。

第4章：介绍了 Python 的内置数据结构——元组、列表、字典和序列。根据使用习惯分别介绍了这些内置数据结构的特点以及区别。

第5章：讲解了 Python 中模块和函数的概念。重点介绍了 Python 的常用内置模块、函数的参数、递归函数、lambda 函数、Generator 函数等内容。

第6章：重点介绍了 Python 中字符串的处理，包括字符串的格式化、比较、合并、截取、查找、替换等。讲解了正则表达式的概念以及使用 re 模块处理正则表达式。

第7章：介绍了 Python 对文件的基本操作，包括文件的创建、读写、删除、复制、搜索、替换和比较。重点介绍了 Python 对目录遍历的实现，以及 Python 的流对象。

第8章：介绍了面向对象程序设计的要点，重点讲解了 Python 如何实现面向对象的特性，以及 Python 中的设计模式。

第9章：详细介绍了 Python 对异常的处理、异常的捕获和抛出、自定义异常等内容。讲解了如何使用 IDLE 和 Easy Eclipse for Python 调试 Python 程序。

第10章：介绍了 Python 的数据库编程，重点讲解了使用 ODBC、DAO、ADO、Python 专用模块连接数据库，以及 Python 的持久化。最后结合 SQLite 数据库的示例讲解了 Python 对数据库的操作。

第11章：介绍 Python 的 GUI 开发。主要介绍了当前 Python 中使用比较多的 Tkinter 和 PyQt。

第12章：介绍了 Python 自带的 GUI 开发库 Tkinter 的基本组件及其使用方法，并给出每种组件的详细示例代码与图示。

第13章：介绍了 Python 的 HTML 应用。详细介绍了 URL 的解析以及 HTML 资源的获取，同时对 CGI 的使用和 HTML 文档的解析进行了介绍。

第14章：讲解了 Python 的 XML 操作，包括 XML 的各种内容实体。还对 XML 文档的两种处理方式 SAX 和 DOM 进行了详细的讲解。

第15章：对 Python 中常用的 Web 开发框架进行了介绍。在介绍了 MVC 模式的基础上，对 Django 框架进行了详细的讲解，包括基本使用方法和高级使用方法。

第16章：介绍了 Python 中的测试框架。主要讲解 Python 中两种测试框架：unittest 和 doctest。讲解的时候，都配以示例，以利于读者掌握。

第17章：介绍了 Python 中进程和线程的概念。主要的内容包括进程和线程的创建及管理。

同时，着重讲解了多线程环境下的数据同步机制。

第 18 章：介绍了 Python 的系统管理，其中主要讲解 IPython 的应用，不仅包括 IPython 的介绍，还包括其常见应用。

第 19 章：讲解了 Python 中和网络编程相关的内容，包括服务器端和客户端的通信。随后着重介绍了网络中的异步通信方式，最后对 Twisted 框架进行了介绍。

第 20 章：讲解了 Python 开发中常见的网络应用。主要包括文件传输、邮件的接收和发送、远程登录以及简单网络管理功能的实现。最后对使用 Scapy 分析网络数据进行了介绍。

第 21 章：讲解了 Python 图像处理，通过各种示例，读者能够迅速掌握相关知识点并用于实践。

第 22 章：介绍了 Python 语言的扩展和嵌入应用。这两种技术使得 Python 语言有了更进一步的发展。

第 23 章：介绍了 Windows 下的 Python 开发，主要介绍了组件对象模型。同时，使用示例来讲解 Windows 下的各种 Python 应用。

适合读者

- Python 爱好者
- 大中专院校的学生
- 社会培训班学生
- 高等教育学校的学生
- 系统管理员
- 界面开发人员
- Web 开发人员
- 网络编程人员

本书作者

本书由张志强、赵越编写，其中天津职业技术师范大学的张志强编写第 1~12 章，渤海大学的赵越编写第 13~23 章，全书最后由范林涛统稿。



目 录

前言

第一篇 Python语言基础

第 1 章 进入 Python 的世界	1
1.1 Python 的由来	1
1.2 Python 的特色	2
1.3 第一个 Python 程序	3
1.4 搭建开发环境	4
1.4.1 Python 的下载和安装	4
1.4.2 交互式命令行的使用	5
1.5 Python 的开发工具	5
1.5.1 PyCharm 的使用	6
1.5.2 Eclipse IDE 的介绍	6
1.5.3 EditPlus 编辑器环境的配置	7
1.6 不同平台下的 Python	9
1.7 小结	9
1.8 习题	9
第 2 章 Python 必须知道的基础语法	10
2.1 Python 的文件类型	10
2.1.1 源代码	10
2.1.2 字节代码	10
2.1.3 优化代码	11
2.2 Python 的编码规范	11
2.2.1 命名规则	11
2.2.2 代码缩进与冒号	13
2.2.3 模块导入的规范	15
2.2.4 使用空行分隔代码	16
2.2.5 正确的注释	16

82.6	2.2.6 语句的分隔	19
82.2	2.3 变量和常量	20
82.6	2.3.1 变量的命名	20
82.2	2.3.2 变量的赋值	21
82.6	2.3.3 局部变量	22
82.6	2.3.4 全局变量	23
80.3	2.3.5 常量	25
82.4	2.4 数据类型	26
82.4	2.4.1 数字	26
82.4	2.4.2 字符串	27
82.5	2.5 运算符与表达式	29
82.5	2.5.1 算术运算符和算术表达式	29
82.5	2.5.2 关系运算符和关系表达式	30
82.5	2.5.3 逻辑运算符和逻辑表达式	31
82.6	2.6 小结	32
82.7	2.7 习题	32
83.0	第3章 Python 的控制语句	33
83.1	3.1 结构化程序设计	33
83.2	3.2 条件判断语句	34
83.2	3.2.1 if 条件语句	34
83.2	3.2.2 if...elif...else 判断语句	35
83.3	3.2.3 if 语句也可以嵌套	36
83.3	3.2.4 switch 语句的替代方案	37
83.3	3.3 循环语句	40
83.3	3.3.1 while 循环	40
83.4	3.3.2 for 循环	41
83.5	3.3.3 break 和 continue 语句	42
83.4	3.4 结构化程序示例	44
83.5	3.5 小结	46
83.6	3.6 习题	46
84.0	第4章 Python 数据结构	47
84.1	4.1 元组结构	47
84.1	4.1.1 元组的创建	47
84.2	4.1.2 元组的访问	48
84.3	4.1.3 元组的遍历	50
84.2	4.2 列表结构	51
84.2	4.2.1 列表的创建	51

4.2.2	列表的使用	53
4.2.3	列表的查找、排序、反转	54
4.2.4	列表实现堆栈和队列	55
4.3	字典结构	57
4.3.1	字典的创建	57
4.3.2	字典的访问	58
4.3.3	字典的方法	60
4.3.4	字典的排序、复制	63
4.3.5	全局字典——sys.modules 模块	64
4.4	序列	65
4.5	小结	67
4.6	习题	67
第 5 章	模块与函数	68
5.1	Python 程序的结构	68
5.2	模块	69
5.2.1	模块的创建	69
5.2.2	模块的导入	69
5.2.3	模块的属性	71
5.2.4	模块的内置函数	71
5.2.5	自定义包	74
5.3	函数	76
5.3.1	函数的定义	76
5.3.2	函数的参数	77
5.3.3	函数的返回值	80
5.3.4	函数的嵌套	82
5.3.5	递归函数	84
5.3.6	lambda 函数	85
5.3.7	Generator 函数	86
5.4	小结	88
5.5	习题	88
第 6 章	字符串与正则表达式	89
6.1	常见的字符串操作	89
6.1.1	字符串的格式化	89
6.1.2	字符串的转义符	91
6.1.3	字符串的合并	93
6.1.4	字符串的截取	94
6.1.5	字符串的比较	95

6.1.6	字符串的反转	96
6.1.7	字符串的查找和替换	98
6.1.8	字符串与日期的转换	99
6.2	正则表达式应用	101
6.2.1	正则表达式简介	101
6.2.2	使用 re 模块处理正则表达式	103
6.3	小结	108
6.4	习题	108
第 7 章	使用 Python 处理文件	109
7.1	文件的常见操作	109
7.1.1	文件的创建	109
7.1.2	文件的读取	111
7.1.3	文件的写入	113
7.1.4	文件的删除	114
7.1.5	文件的复制	115
7.1.6	文件的重命名	116
7.1.7	文件内容的搜索和替换	117
7.1.8	文件的比较	118
7.1.9	配置文件的访问	119
7.2	目录的常见操作	121
7.2.1	创建和删除目录	122
7.2.2	目录的遍历	122
7.3	文件和流	124
7.3.1	Python 的流对象	124
7.3.2	模拟 Java 的输入、输出流	126
7.4	文件处理示例——文件属性浏览程序	127
7.5	小结	128
7.6	习题	128
第 8 章	面向对象编程	129
8.1	面向对象的概述	129
8.2	类和对象	130
8.2.1	类和对象的区别	130
8.2.2	类的定义	131
8.2.3	对象的创建	131
8.3	属性和方法	132
8.3.1	类的属性	132
8.3.2	类的方法	134

8.3.3	内部类的使用	136
8.3.4	<code>__init__</code> 方法	137
8.3.5	<code>__del__</code> 方法	138
8.3.6	垃圾回收机制	139
8.3.7	类的内置方法	140
8.3.8	方法的动态特性	144
8.4	继承	145
8.4.1	使用继承	145
8.4.2	抽象基类	147
8.4.3	多态性	148
8.4.4	多重继承	149
8.4.5	Mixin 机制	150
8.5	运算符的重载	152
8.6	Python 与设计模式	154
8.6.1	设计模式简介	154
8.6.2	设计模式示例——Python 实现工厂方法	155
8.7	小结	156
8.8	习题	156
第 9 章	异常处理与程序调试	157
9.1	异常的处理	157
9.1.1	Python 中的异常	157
9.1.2	<code>try...except</code> 的使用	158
9.1.3	<code>try...finally</code> 的使用	160
9.1.4	使用 <code>raise</code> 抛出异常	161
9.1.5	自定义异常	161
9.1.6	<code>assert</code> 语句的使用	162
9.1.7	异常信息	163
9.2	使用自带 IDLE 调试程序	164
9.3	使用 Easy Eclipse for Python 调试程序	165
9.3.1	新建工程	166
9.3.2	配置调试	167
9.3.3	设置断点	168
9.4	小结	170
9.5	习题	170
第 10 章	Python 数据库编程	171
10.1	Python 环境下的数据库编程	171
10.1.1	通过 ODBC 访问数据库	171

10.1.2	使用 DAO 对象访问数据库	173
10.1.3	使用 ActiveX Data Object 访问数据库	174
10.1.4	Python 连接数据库的专用模块	176
10.2	使用 Python 的持久化模块读写数据	179
10.3	嵌入式数据库 SQLite	179
10.3.1	SQLite 的命令行工具	179
10.3.2	使用 sqlite3 模块访问 SQLite 数据库	181
10.4	小结	182
10.5	习题	182

第二篇 Python的GUI程序设计

第 11 章	Python 的 GUI 开发	183
11.1	Python 的 GUI 开发选择	183
11.1.1	认识 Python 内置的 GUI 库 Tkinter	183
11.1.2	使用 Tkinter 进行开发	184
11.1.3	认识 PyQt GUI 库	186
11.1.4	使用 PyQt GUI 库进行开发	187
11.2	小结	188
11.3	习题	188
第 12 章	GUI 编程与 Tkinter 相关组件介绍	189
12.1	GUI 程序开发简介	189
12.2	Tkinter 与主要组件	190
12.2.1	在程序中使用 Tkinter	190
12.2.2	顶层窗口	190
12.2.3	标签	191
12.2.4	框架	191
12.2.5	按钮	192
12.2.6	输入框	192
12.2.7	单选按钮	193
12.2.8	复选按钮	193
12.2.9	消息	194
12.2.10	滚动条	194
12.2.11	列表框	195
12.3	Tkinter 所有组件简介	195
12.4	小结	196
12.5	习题	196

第三篇 Python的Web开发

第 13 章 Python 的 HTML 应用	197
13.1 HTML 介绍	197
13.1.1 HTML 的历史	197
13.1.2 SGML、HTML、XHTML、HTML5 的关系	198
13.1.3 HTML 的标签	198
13.1.4 HTML 的框架组成	199
13.2 URL 的处理	200
13.2.1 统一资源定位符 URL	200
13.2.2 URL 的解析	201
13.2.3 URL 的拼合	203
13.2.4 URL 的分解	204
13.2.5 URL 的编解码	205
13.2.6 中文的编解码	207
13.2.7 查询参数的编码	208
13.3 CGI 的使用	210
13.3.1 CGI 介绍	210
13.3.2 获取 CGI 环境信息	211
13.3.3 解析用户的输入	214
13.4 获取 HTML 资源	216
13.4.1 使用 urlopen 和 urlretrieve 获取 HTTP 资源	217
13.4.2 分析返回资源的相关信息	221
13.4.3 自定义获取资源方式	223
13.4.4 使用 http.client 模块获取资源	226
13.5 HTML 文档的解析	228
13.6 小结	230
13.7 习题	230
第 14 章 Python 和 XML	231
14.1 XML 介绍	231
14.1.1 XML 的演进历史	231
14.1.2 XML 的优点和限制	232
14.1.3 XML 技术的 Python 支持	233
14.2 XML 文档概览和验证	234
14.2.1 XML 文档的基础概念	234
14.2.2 XML 文档的结构良好性验证	234
14.2.3 XML 文档的有效性验证	237
14.3 分析 XML 文档结构	239

14.3.1	XML 的元素和标签	239
14.3.2	元素的属性	242
14.3.3	XML 的名字	244
14.3.4	字符实体	245
14.3.5	CDATA 段	246
14.3.6	注释	248
14.3.7	处理指令	248
14.3.8	XML 定义	249
14.4	使用 SAX 处理 XML 文档	249
14.4.1	SAX 介绍	250
14.4.2	SAX 处理的组成部分	250
14.5	使用 DOM 处理 XML 文档	255
14.5.1	DOM 介绍	255
14.5.2	xml.dom 模块中的接口操作	256
14.5.3	对 XML 文档的操作	264
14.6	小结	269
14.7	习题	270
第 15 章	Python 的 Web 开发——Django 框架的应用	271
15.1	常见的 Web 开发框架	271
15.1.1	Zope	271
15.1.2	TurboGears	273
15.1.3	Django	273
15.1.4	其他 Web 开发框架	274
15.1.5	根据自身所需选择合适的开发框架	275
15.2	MVC 模式	275
15.2.1	MVC 模式介绍	275
15.2.2	MVC 模式的优缺点	276
15.2.3	Django 框架中的 MVC	277
15.3	Django 开发环境的搭建	277
15.3.1	Django 框架的安装	277
15.3.2	数据库的配置	278
15.4	Django 框架的应用	279
15.4.1	Web 应用的创建	279
15.4.2	Django 中的开发服务器	280
15.4.3	创建数据库	282
15.4.4	生成 Django 应用	283
15.4.5	创建数据模型	284

15.4.6	URL 设计	285
15.4.7	创建视图	286
15.4.8	模板系统	287
15.4.9	发布 Django 项目	289
15.5	Django 框架的高级应用	289
15.5.1	管理界面	290
15.5.2	生成数据库数据	291
15.5.3	Session 功能	292
15.5.4	国际化	295
15.6	小结	296
15.7	习题	296

第四篇 Python其他应用

第 16 章	敏捷方法学在 Python 中的应用——测试驱动开发	297
16.1	测试驱动开发	297
16.1.1	测试驱动开发模式	297
16.1.2	TDD 的优势	298
16.1.3	TDD 的使用步骤	299
16.2	unittest 测试框架	300
16.2.1	unittest 模块介绍	300
16.2.2	构建测试用例	301
16.2.3	构建测试固件	302
16.2.4	组织多个测试用例	304
16.2.5	构建测试套件	305
16.2.6	重构代码	307
16.2.7	执行测试	308
16.3	使用 doctest 进行测试	311
16.3.1	doctest 模块介绍	311
16.3.2	构建可执行文档	312
16.3.3	执行 doctest 测试	313
16.4	小结	315
16.5	习题	315
第 17 章	Python 中的进程和线程	316
17.1	进程和线程	316
17.1.1	进程和线程的概念	316
17.1.2	Python 中对于进程和线程处理的支持	317
17.2	Python 下的进程编程	318
17.2.1	进程运行环境	318

17.2.2	创建进程	319
17.2.3	终止进程	320
17.3	使用 subprocess 模块管理进程	320
17.3.1	使用 Popen 类管理进程	321
17.3.2	调用外部系统命令	324
17.3.3	替代其他进程创建函数	324
17.4	进程间的信号机制	325
17.4.1	信号的处理	325
17.4.2	信号使用的规则	327
17.5	多线程概述	328
17.5.1	什么是多线程	328
17.5.2	线程的状态	328
17.5.3	Python 中的线程支持	329
17.6	生成和终止线程	329
17.6.1	使用_thread 模块	329
17.6.2	使用 threading.Thread 类	332
17.7	管理线程	334
17.7.1	线程状态转移	334
17.7.2	主线程对子线程的控制	334
17.7.3	线程中的局部变量	335
17.8	线程之间的同步	336
17.8.1	临界资源和临界区	336
17.8.2	锁机制	338
17.8.3	条件变量	339
17.8.4	信号量	342
17.8.5	同步队列	342
17.8.6	线程同步小结	344
17.9	小结	344
17.10	习题	344
第 18 章	基于 Python 的系统管理	345
18.1	增强的交互式环境 IPython	345
18.1.1	IPython 介绍	345
18.1.2	IPython 的安装	346
18.1.3	IPython 的启动	347
18.1.4	IPython 的环境配置	348
18.2	和 IPython 的简单交互	349
18.2.1	IPython 中的输入和输出	349
18.2.2	输出提示符的区别	349

18.2.3	输出提示符区别的原因	350
18.3	IPython 中的 magic 函数	352
18.3.1	magic 函数的使用和构造	352
18.3.2	目录管理	352
18.3.3	对象信息的收集	356
18.3.4	magic 函数小结	359
18.4	IPython 适合于系统管理的特点	359
18.4.1	Tab 补全	359
18.4.2	历史记录功能	361
18.4.3	执行外部系统命令和运行文件	363
18.4.4	对象查看和自省	367
18.4.5	直接编辑代码	370
18.4.6	设置别名和宏	371
18.5	使用 Python 进行文件管理	373
18.5.1	文件的比较	373
18.5.2	文件的归档	375
18.5.3	文件的压缩	377
18.6	使用 Python 定时执行任务	379
18.6.1	使用休眠功能	379
18.6.2	使用 sched 模块来定时执行任务	380
18.7	小结	380
18.8	习题	380
第 19 章	Python 和网络编程	381
19.1	网络模型介绍	381
19.1.1	OSI 简介	381
19.1.2	TCP/IP 简介	382
19.2	Socket 应用	383
19.2.1	Socket 基础	383
19.2.2	Socket 的工作方式	383
19.3	服务器端和客户端通信	384
19.3.1	服务器端的构建	384
19.3.2	客户端的构建	389
19.4	异步通信方式	391
19.4.1	使用 Fork 方式	391
19.4.2	使用线程方式	392
19.4.3	使用异步 IO 方式	393
19.4.4	使用 asyncore 模块	397