



高等学校计算机科学与技术教材

Python 程序设计教程

COMPUTER Science and Technology

□ 江红 余青松 编著

- 原理与技术的完美结合
- 教学与科研的最新成果
- 语言精练，实例丰富
- 可操作性强，实用性突出



清华大学出版社 ● 北京交通大学出版社

高等学校计算机科学与技术教材

Python 程序设计教程

江 红 余青松 编著

清华大学出版社
北京交通大学出版社
· 北京 ·

内 容 简 介

本书基于 Windows 7 和 Python 3.3.2 构建 Python 开发平台, 阐述 Python 语言的基础知识, 以及使用 Python 语言的实际开发应用实例。本书集教材、练习册、上机指导于一体, 具体内容包括: Python 概述、Python 语言基础、程序流程控制、数值类型、系列(元组、列表和字符串)、字典和集合类型、文件和流 I/O、函数与函数编程、类和对象、模块和包、迭代器和生成器、数据结构和算法、日期和时间处理、正则表达式、多线程编程、图形用户界面应用程序、数据库访问、网络编程和通信、系统管理。

本书作者结合多年的程序设计、开发及授课经验, 由浅入深、循序渐进地介绍 Python 程序设计语言, 让读者能够较为系统全面地掌握程序设计的理论和应用。

本书可作为高等学校各专业的计算机程序设计教材, 同时也可作为广大程序设计开发者、爱好者的自学参考书。

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 13501256678 13801310933

图书在版编目(CIP)数据

Python 程序设计教程 / 江红, 余青松编著. —北京: 北京交通大学出版社: 清华大学出版社, 2014. 9

(高等学校计算机科学与技术教材)

ISBN 978-7-5121-2043-3

I. ①P… II. ①江… ②余… III. ①软件工具-程序设计-高等学校-教材
IV. ①TP311.56

中国版本图书馆 CIP 数据核字(2014)第 188247 号

责任编辑: 谭文芳

出版发行: 清华大学出版社 邮编: 100084 电话: 010-62776969

北京交通大学出版社 邮编: 100044 电话: 010-51686414

印刷者: 北京交大印刷厂

经 销: 全国新华书店

开 本: 185 × 260 印张: 23.75 字数: 729 千字

版 次: 2014 年 9 月第 1 版 2014 年 9 月第 1 次印刷

书 号: ISBN 978-7-5121-2043-3/TP · 793

印 数: 1 ~ 2 000 册 定价: 45.00 元

本书如有质量问题, 请向北京交通大学出版社质监局反映。对您的意见和批评, 我们表示欢迎和感谢。

投诉电话: 010-51686043, 51686008; 传真: 010-62225406; E-mail: press@bjtu.edu.cn。

前 言

程序设计是大专院校计算机、电子信息、工商管理等相关专业的必修课程。Python 语言是一种解释型、面向对象的计算机程序设计语言，广泛用于计算机程序设计教学语言、系统管理编程脚本语言、科学计算等，特别适用于快速的应用程序开发。Python 编程语言广受开发者的喜爱，并被列入 LAMP（Linux、Apache、MySQL 以及 Python/Perl/PHP）。

本教程集“教材、练习册、上机指导”于一体，基于 Windows 7 和 Python 3.3.2 构建 Python 开发平台，阐述 Python 语言的基础知识，以及使用 Python 语言的实际开发应用实例，具体内容包括：Python 概述、Python 语言基础、程序流程控制、数值类型、系列（元组、列表和字符串）、字典和集合类型、文件和流 I/O、函数与函数编程、类和对象、模块和包、迭代器和生成器、数据结构和算法、日期和时间处理、正则表达式、多线程编程、图形用户界面应用程序、数据库访问、网络编程和通信、系统管理。

本教程涉及的各章节所有的源程序代码和相关素材，以及供教师参考的教学电子文稿均可以通过北京交通大学出版社网站 <http://www.bjtup.com.cn> 下载，也可以通过电子邮件 hjiang@cc.ecnu.edu.cn 直接与作者联系。

本教程由华东师范大学江红和余青松共同编写。衷心感谢本书的编辑谭文芳老师，敬佩她的睿智和敬业。由于时间和编者学识有限，书中不足之处在所难免，敬请诸位同行、专家和读者指正。

编 者
2014 年 7 月

目 录

第 1 章 Python 概述	1
1.1 Python 语言概述	1
1.1.1 Python 语言简介	1
1.1.2 Python 语言的特点	1
1.1.3 Python 语言的应用范围	2
1.2 Python 语言的版本和开发环境	2
1.2.1 Python 语言的版本	2
1.2.2 Python 语言的实现	3
1.2.3 Python 语言的集成开发环境	3
1.3 下载和安装 Python	3
1.3.1 下载 Python	3
1.3.2 安装 Python	4
1.4 使用 Python 解释器解释执行 Python 程序	5
1.4.1 运行 Python 解释器	5
1.4.2 运行 Python 集成开发环境	6
1.5 使用文本编辑器和命令行编写和执行 Python 源文件程序	7
1.5.1 编写 Hello world 程序	7
1.5.2 Hello world 程序 (hello.py) 源代码分析	8
1.5.3 运行 Python 源代码程序	8
1.6 使用集成开发环境 IDLE 编写和执行 Python 源文件程序	9
1.6.1 使用 IDLE 编写 Hello world 程序	10
1.6.2 使用 IDLE 编辑程序	10
1.7 复习题	11
1.8 上机实践	12
第 2 章 Python 语言基础	13
2.1 语句	13
2.1.1 Python 语句	13
2.1.2 Python 语句的书写规则	13
2.1.3 复合语句及其缩进书写规则	14
2.1.4 注释语句	14
2.1.5 空语句 pass	15
2.2 表达式	15
2.2.1 表达式的组成	15

2.2.2	表达式的书写规则	15
2.3	运算符	15
2.3.1	运算符概述	15
2.3.2	Python 运算符及其优先级	16
2.4	标识符及其命名规则	17
2.4.1	标识符	17
2.4.2	保留关键字	17
2.4.3	Python 预定义标识符	18
2.4.4	命名规则	18
2.5	对象与引用	18
2.5.1	对象的类型 (type) 和标识 (id)	18
2.5.2	对象引用	19
2.5.3	对象比较 (==) 和类型判别 (is)	19
2.5.4	不可变对象 (immutable) 和可变对象 (mutable)	20
2.6	变量和赋值	20
2.6.1	变量和数据类型	20
2.6.2	变量的声明和赋值	21
2.6.3	链式赋值语句	21
2.6.4	复合赋值语句	21
2.6.5	删除变量 (del)	22
2.6.6	系列解包赋值	22
2.7	数据类型	23
2.7.1	NoneType, NotImplementedType 和 Ellipsis	23
2.7.2	数值数据类型	23
2.7.3	序列数据类型	24
2.7.4	集合数据类型	25
2.7.5	字典数据类型	25
2.7.6	其他数据类型	25
2.8	类的声明和对象的创建与调用	26
2.8.1	类的声明	26
2.8.2	对象的创建和调用	26
2.9	函数	27
2.9.1	函数的声明和调用	27
2.9.2	内置函数	28
2.9.3	模块函数和 import 语句	28
2.9.4	输入和输出函数	29
2.9.5	运行时提示输入密码	29
2.10	模块和包	30
2.11	Python 文档注释	30

2.11.1	文档字符串	30
2.11.2	文档注释规范	31
2.12	复习题	32
2.13	上机实践	35
第3章	程序流程控制	36
3.1	bool 数据类型和相关运算符	36
3.1.1	bool 类型	36
3.1.2	关系和测试运算符	36
3.1.3	逻辑运算符	37
3.2	顺序结构	38
3.3	选择结构	39
3.3.1	分支结构的形式	39
3.3.2	单分支结构	39
3.3.3	双分支结构	40
3.3.4	多分支结构	41
3.3.5	if 语句的嵌套	42
3.3.6	选择结构综合举例	43
3.4	循环结构	44
3.4.1	可迭代对象 (iterable)	44
3.4.2	for 循环	45
3.4.3	range 对象	45
3.4.4	while 循环	46
3.4.5	循环的嵌套	47
3.4.6	break 语句	48
3.4.7	continue 语句	49
3.4.8	else 子句	50
3.5	异常处理	51
3.5.1	错误和异常	51
3.5.2	异常处理概述	51
3.5.3	内置的异常类	52
3.5.4	自定义异常类	53
3.5.5	引发异常	53
3.5.6	捕获处理异常 try...except...finally	54
3.6	断言处理	56
3.6.1	断言处理概述	56
3.6.2	assert 语句和 AssertionError 类	57
3.6.3	启用/禁用断言	57
3.7	复习题	58
3.8	上机实践	62

第 4 章 数值类型	64
4.1 int 类型（任意精度整数）	64
4.1.1 整型常量	64
4.1.2 int 对象	65
4.1.3 数制转换函数	65
4.1.4 整数的运算	65
4.2 float 类型（有限精度浮点数）	65
4.2.1 浮点类型常量	66
4.2.2 float 对象	66
4.2.3 浮点数的运算	66
4.3 Decimal 类型（高精度浮点数）	67
4.3.1 浮点数运算误差	67
4.3.2 Decimal 对象	67
4.3.3 Context 对象	68
4.3.4 高精度浮点数的运算	71
4.4 Fraction 类型（分数）	71
4.4.1 Fraction 对象	71
4.4.2 分数的运算	72
4.5 complex 类型（复数）	72
4.5.1 复数类型常量	72
4.5.2 创建 complex 对象	72
4.5.3 complex 类型的方法	73
4.5.4 复数的运算	73
4.6 算术运算符和位运算符	73
4.6.1 算术运算符	73
4.6.2 位运算符	74
4.6.3 混合运算和数值类型转换	74
4.6.4 内置标准数学函数	75
4.7 math 模块和数学函数	75
4.8 cmath 模块和复数数学函数	78
4.9 random 模块和随机函数	79
4.9.1 种子和随机状态	80
4.9.2 随机整数	80
4.9.3 随机系列	81
4.9.4 随机实值分布	82
4.10 相关模块	82
4.10.1 标准库中的相关模块	82
4.10.2 数值运算模块 NumPy	83
4.10.3 科学计算模块 SciPy	83

4.11 复习题	83
4.12 上机实践	84
第5章 系列：元组、列表和字符串	89
5.1 系列的基本操作	89
5.1.1 系列的索引访问操作	89
5.1.2 系列的切片操作	90
5.1.3 系列的连接和重复操作	91
5.1.4 系列的成员关系操作	91
5.1.5 系列的比较运算操作	92
5.1.6 系列的排序操作	92
5.1.7 系列长度、最大值、最小值、求和	93
5.1.8 内置函数 all() 和 any()	93
5.1.9 系列拆封	93
5.2 元组	94
5.2.1 元组的定义	94
5.2.2 元组的基本操作	95
5.3 列表	95
5.3.1 列表的定义	95
5.3.2 列表的基本操作	95
5.3.3 list 对象的方法	96
5.3.4 列表解析表达式	96
5.3.5 列表作为队列和栈	97
5.4 字符串	97
5.4.1 Unicode 和字符常量	97
5.4.2 字符串常量	98
5.4.3 创建 str 对象	99
5.4.4 字符串的基本操作	99
5.4.5 str 对象的方法	99
5.4.6 字符串编码	102
5.4.7 字符串格式化	103
5.5 字节系列	105
5.5.1 bytes 常量	105
5.5.2 创建 bytes 对象	106
5.5.3 创建 bytearray 对象	106
5.5.4 bytes 和 bytearray 的基本操作	107
5.6 复习题	107
5.7 上机实践	110
第6章 字典和集合类型	112
6.1 字典	112

6.1.1	对象的 hash 值	112
6.1.2	字典的定义	112
6.1.3	字典的访问操作	113
6.1.4	字典的视图对象	113
6.1.5	判断字典键是否存在	113
6.1.6	字典对象的长度和比较	114
6.1.7	字典对象的方法	114
6.2	集合	115
6.2.1	集合的定义	115
6.2.2	判断集合元素是否存在	115
6.2.3	集合的运算：并集、交集、差集和对称差集	116
6.2.4	集合的比较运算：相等、子集和超集	116
6.2.5	集合的长度、最大值、最小值、元素和	117
6.2.6	可变集合的方法	117
6.3	复习题	118
6.4	上机实践	119
第 7 章	文件和流 I/O	121
7.1	文件和文件对象	121
7.1.1	文件和流概述	121
7.1.2	文件对象和 open 函数	121
7.1.3	with 语句和上下文管理协议	122
7.2	文本文件的读取和写入	122
7.2.1	文本文件的写入	122
7.2.2	文本文件的读取	123
7.2.3	文本文件的编码	124
7.3	二进制文件的读取和写入	125
7.3.1	二进制文件的写入	125
7.3.2	二进制文件的读取	125
7.4	随机文件访问	126
7.5	CSV 文件格式的读取和写入	127
7.5.1	csv.reader 对象和 csv 文件的读取	127
7.5.2	csv.writer 对象和 csv 文件的写入	128
7.5.3	csv.DictReader 对象和 csv 文件的读取	128
7.5.4	csv.DictWriter 对象和 csv 文件的写入	129
7.5.5	csv 文件格式化参数和 Dialect 对象	130
7.6	os 模块和文件访问	131
7.6.1	文件描述符	131
7.6.2	使用 os 模块提供的函数访问文件	131
7.7	对象序列化	132

7.7.1	对象序列化	132
7.7.2	pickle 模块和对象序列化	132
7.8	输入重定向和管道	133
7.8.1	FileInput 对象	133
7.8.2	fileinput 模块的函数	134
7.8.3	输入重定向	135
7.9	复习题	136
7.10	上机实践	136
第 8 章	函数与函数编程	137
8.1	函数的声明和调用	137
8.2	参数的传递	138
8.2.1	形式参数和实际参数	138
8.2.2	可选参数	138
8.2.3	位置参数和命名参数	139
8.2.4	可变参数 (VarArgs)	140
8.2.5	强制命名参数 (Keyword - only)	140
8.3	函数的返回值	141
8.4	变量的作用域	142
8.4.1	全局变量	142
8.4.2	局部变量	142
8.4.3	类成员变量	142
8.4.4	全局语句 global	142
8.4.5	非局部语句 nonlocal	143
8.5	函数装饰器	144
8.5.1	装饰器的声明和使用	144
8.5.2	@functools.wraps 装饰器	146
8.5.3	functools.update_wrapper 函数	146
8.6	递归函数	147
8.7	作为对象的函数	148
8.8	Lambda 表达式和匿名函数	148
8.9	operator 模块和运算符函数	149
8.10	eval、exec 和 compile 函数	149
8.10.1	运行上下文的局部变量和全局变量	149
8.10.2	eval 函数和动态表达式的求值	150
8.10.3	exec 函数和动态语句的执行	150
8.10.4	compile 函数和动态语句的执行	150
8.11	functools 模块和函数工具	150
8.11.1	partial 对象	151
8.11.2	reduce 函数	151

8.11.3	@functools.lru_cache 装饰器	151
8.12	复习题	152
8.13	上机实践	153
第9章	类和对象	155
9.1	面向对象概念	155
9.1.1	对象的定义	155
9.1.2	封装	155
9.1.3	继承	156
9.1.4	多态性	156
9.2	类和对象	156
9.2.1	类的声明	157
9.2.2	对象的创建和使用	157
9.3	属性	157
9.3.1	实例属性和类属性	158
9.3.2	私有属性和公有属性	160
9.3.3	@property 装饰器	160
9.3.4	特殊属性 (Special Attributes)	161
9.3.5	自定义属性 (Custom Attributes)	162
9.4	方法	163
9.4.1	方法的声明和调用	163
9.4.2	__init__方法(构造函数)和__new__方法	164
9.4.3	__del__方法(析构函数)	164
9.4.4	实例方法、类方法(@classmethod)与静态方法(@staticmethod)	165
9.4.5	私有方法与公有方法	166
9.4.6	方法重载	167
9.4.7	运算符重载	168
9.4.8	@functools.total_ordering 装饰器	169
9.4.9	特殊方法(Special Method)	170
9.4.10	__call__方法和可调用对象(callable)	170
9.5	继承	171
9.5.1	派生类	171
9.5.2	类成员的继承和重写	172
9.6	对象的引用、浅拷贝和深拷贝	173
9.6.1	对象的引用	173
9.6.2	对象的浅拷贝	173
9.6.3	对象的深拷贝	174
9.7	复习题	174
9.8	上机实践	175
第10章	模块和包	177

10.1	模块的导入和使用	177
10.1.1	import 语句	177
10.1.2	from...import 语句	177
10.1.3	__import__() 内置函数	178
10.1.4	模块搜索路径 sys.path	178
10.1.5	dir() 内置函数	178
10.2	模块的定义	179
10.2.1	创建模块	179
10.2.2	模块的名称	180
10.2.3	按字节编译的 .pyc 文件	181
10.3	包的定义	181
10.3.1	包的概念	181
10.3.2	创建包	182
10.3.3	包的导入和使用	182
10.4	命令行参数	182
10.4.1	sys.argv 与命令行参数	182
10.4.2	argparse 和命令行选项参数解析	183
10.5	终止程序运行时返回消息	185
10.6	复习题	185
10.7	上机实践	186
第 11 章	迭代器和生成器	187
11.1	迭代和可迭代对象	187
11.1.1	可迭代对象、迭代器和可迭代协议	187
11.1.2	可迭代对象的迭代: iter 和 next 函数	188
11.1.3	可迭代对象的迭代: for 语句	188
11.2	自定义可迭代对象和迭代器	189
11.3	生成器函数	189
11.4	反向迭代: reversed 迭代器	190
11.5	生成器表达式	191
11.6	Python 内置的可迭代对象	191
11.6.1	range 可迭代对象	191
11.6.2	map 迭代器和 itertools.starmap 迭代器	191
11.6.3	filter 迭代器和 itertools.filterfalse 迭代器	192
11.6.4	zip 迭代器和 zip_longest 迭代器	192
11.6.5	enumerate 迭代器	193
11.7	itertools 模块和迭代器函数	193
11.7.1	无穷系列迭代器	193
11.7.2	累计迭代器 accumulate	194
11.7.3	级联迭代器 chain	194

11.7.4	选择压缩迭代器 <code>compress</code>	194
11.7.5	截取迭代器 <code>dropwhile</code> 和 <code>takewhile</code>	194
11.7.6	切片迭代器 <code>islice</code>	195
11.7.7	迭代器 <code>groupby</code>	195
11.7.8	返回多个迭代器 <code>tee</code>	195
11.7.9	组合迭代器 <code>combinations</code> 、 <code>combinations_with_replacement</code>	195
11.7.10	排列迭代器 <code>permutations</code>	196
11.7.11	笛卡儿积迭代器 <code>product</code>	196
11.8	复习题	196
11.9	上机实践	198
第 12 章	数据结构和算法	200
12.1	ABC 模块	200
12.1.1	<code>collections.abc</code> 模块和抽象基类	200
12.1.2	<code>collections.abc</code> 模块中的抽象基类	200
12.2	<code>collections</code> 模块和容器类型	202
12.2.1	<code>ChainMap</code> 对象	202
12.2.2	<code>Counter</code> 对象	203
12.2.3	<code>deque</code> 对象	203
12.2.4	<code>defaultdict</code> 对象	205
12.2.5	<code>OrderedDict</code> 对象	205
12.2.6	<code>namedtuple</code> 对象	206
12.2.7	<code>UserDict</code> 、 <code>UserList</code> 和 <code>UserString</code> 对象	207
12.3	<code>array.array</code> 对象	207
12.3.1	<code>array</code> 的定义	207
12.3.2	<code>array</code> 对象的基本操作	208
12.3.3	<code>array</code> 对象的方法	208
12.4	<code>heapq</code> 模块和堆队列算法	209
12.4.1	堆 (Heap) 的基本概念	209
12.4.2	堆的基本操作	210
12.4.3	<code>heapq</code> 模块中基于 <code>heap</code> 的通用函数	210
12.5	<code>bisect</code> 模块和二分排序算法	211
12.6	复习题	211
12.7	上机实践	213
第 13 章	日期和时间处理	214
13.1	日期和时间处理概述	214
13.1.1	相关模块	214
13.1.2	相关术语	214
13.2	<code>time</code> 模块	215
13.2.1	<code>struct_time</code> 对象	215

13.2.2	time 模块中的常用函数	215
13.2.3	日期格式化字符串	216
13.3	datetime 模块	217
13.3.1	date 对象	218
13.3.2	tzinfo 对象和 timezone 对象	219
13.3.3	time 对象	220
13.3.4	datetime 对象	220
13.3.5	timedelta 对象	222
13.3.6	日期和时间的运算与比较	223
13.4	calendar 模块	223
13.4.1	Calendar 对象	223
13.4.2	TextCalendar 对象和 LocaleTextCalendar 对象	224
13.4.3	HTMLCalendar 对象和 LocaleHTMLCalendar 对象	225
13.4.4	calendar 模块的属性和常用函数	226
13.5	复习题	227
13.6	上机实践	228
第 14 章	正则表达式	230
14.1	正则表达式语言	230
14.1.1	正则表达式语言概述	230
14.1.2	正则表达式引擎	230
14.1.3	普通字符和转义字符	231
14.1.4	字符类和预定义字符类	231
14.1.5	边界匹配符	232
14.1.6	重复限定符	232
14.1.7	匹配算法:贪婪和懒惰	233
14.1.8	选择分支	234
14.1.9	分组和向后引用	234
14.1.10	正则表达式的匹配模式	236
14.1.11	常用正则表达式	236
14.2	正则表达式模块 re	236
14.2.1	匹配和搜索函数	236
14.2.2	匹配选项	237
14.2.3	正则表达式对象	237
14.2.4	匹配对象	238
14.2.5	匹配和替换	238
14.2.6	分割字符串	238
14.3	正则表达式应用举例	239
14.3.1	字符串包含验证	239
14.3.2	字符串查找和拆分	239

14.3.3	字符串替换和清除	240
14.3.4	字符串查找和截取	240
14.4	复习题	241
14.5	上机实践	242
第 15 章	多线程编程	243
15.1	线程处理概述	243
15.1.1	进程和线程	243
15.1.2	线程的优缺点	243
15.2	创建和启动多线程	244
15.2.1	使用 start_new_thread 函数创建线程	244
15.2.2	使用 Thread 对象创建线程	244
15.2.3	自定义派生于 Thread 的对象	245
15.2.4	线程加入 join()	246
15.2.5	用户线程和 daemon 线程	247
15.3	线程同步	248
15.3.1	线程同步处理	248
15.3.2	基于原语锁 (Lock/RLock 对象) 的简单同步	248
15.3.3	基于条件变量 (Condition 对象) 的同步和通信	250
15.4	复习题	253
15.5	上机实践	253
第 16 章	图形用户界面应用程序	254
16.1	图形用户界面概述	254
16.1.1	tkinter	254
16.1.2	其他 GUI 库简介	254
16.2	tkinter 概述	255
16.2.1	tkinter 模块	255
16.2.2	图形用户界面构成	255
16.2.3	框架和 GUI 应用程序类	256
16.2.4	tkinter 主窗口	257
16.3	几何布局管理器	257
16.3.1	pack 几何布局管理器	258
16.3.2	grid 几何布局管理器	259
16.3.3	place 几何布局管理器	260
16.4	事件处理	261
16.4.1	事件类型	261
16.4.2	事件绑定	262
16.4.3	事件处理函数	263
16.5	组件概述	264
16.5.1	创建组件和设置属性	264

16.5.2	坐标系	265
16.5.3	大小单位 (width 和 height)	265
16.5.4	颜色 (background、foreground 等)	265
16.5.5	字体 (font)	266
16.5.6	停靠位置 (anchor)	266
16.5.7	光标 (cursor)	267
16.5.8	显示文本 (text、textvariable、wraplength 和 justify)	267
16.5.9	位图 (bitmap)	267
16.5.10	图像 (image)	268
16.5.11	同时显示文本和位图/图像 (compound)	268
16.5.12	3D 显示样式 (relief 和 overrelief)	269
16.5.13	设置组件的边框 (borderwidth)	269
16.5.14	设置组件的填充 (padx 和 pady)	269
16.5.15	组件的状态 (state)	269
16.5.16	启用和禁用输入法	269
16.5.17	设置字符下划线位置 (underline)	269
16.5.18	焦点 (highlightbackground、highlightcolor 等)	270
16.5.19	组件的数据绑定 (textvariable、variable 和 listvariable)	270
16.5.20	组件 id 和名称	270
16.5.21	组件的通用方法	270
16.6	常用组件	271
16.6.1	标签 Label	271
16.6.2	标签框架 LabelFrame	271
16.6.3	按钮 Button	272
16.6.4	消息 Message	273
16.6.5	单行文本框 Entry	274
16.6.6	多行文本框 Text	275
16.6.7	单选按钮 Radiobutton	277
16.6.8	复选框 Checkbutton	278
16.6.9	列表框 Listbox	280
16.6.10	选择项 OptionMenu	282
16.6.11	移动滑块 Scale	283
16.6.12	顶层窗口 Toplevel	285
16.6.13	ttk 子模块控件	286
16.7	对话框	286
16.7.1	通用消息对话框	286
16.7.2	文件对话框	288
16.7.3	颜色选择对话框	289
16.7.4	通用对话框应用举例	290