

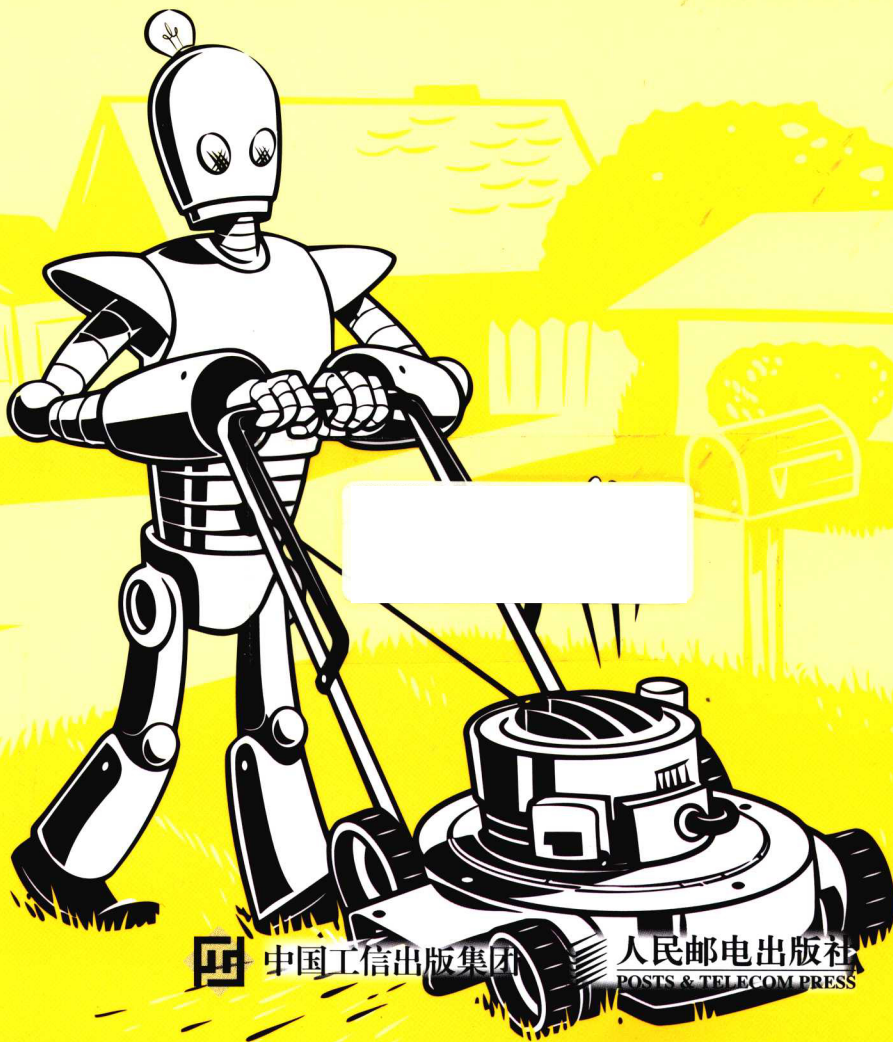
资深Python程序员力作 带你快速掌握Python高效编程



Python编程快速上手 ——让繁琐工作自动化

AUTOMATE THE BORING STUFF WITH PYTHON

[美] Al Sweigart 著 王海鹏 译



中国工信出版集团

人民邮电出版社
POSTS & TELECOM PRESS



Python编程快速上手 ——让繁琐工作自动化

AUTOMATE THE BORING STUFF WITH PYTHON

[美] Al Sweigart 著 王海鹏 译

人民邮电出版社

北京

图书在版编目 (C I P) 数据

Python编程快速上手：让繁琐工作自动化 / (美)
斯维加特 (Al Sweigart) 著；王海鹏译. — 北京：人
民邮电出版社，2016.7
ISBN 978-7-115-42269-9

I. ①P… II. ①斯… ②王… III. ①软件工具—程序
设计 IV. ①TP311.56

中国版本图书馆CIP数据核字(2016)第110206号

版权声明

Simplified Chinese-language edition copyright © 2016 by Posts and Telecom Press.

Copyright © 2015 by Al Sweigart. Title of English-language original: Automate The Boring Stuff with Python ISBN-13: 978-1-59327-599-0, published by No Starch Press.

All rights reserved.

本书中文简体字版由美国 No Starch 出版社授权人民邮电出版社出版。未经出版者书面许可，对本书任何部分不得以任何方式复制或抄袭。

版权所有，侵权必究。

-
- ◆ 著 [美] Al Sweigart
 - 译 王海鹏
 - 责任编辑 陈冀康
 - 责任印制 焦志炜

 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京鑫正大印刷有限公司印刷

 - ◆ 开本：800×1000 1/16
印张：26.25
字数：590 千字 2016 年 7 月第 1 版
印数：1—3 000 册 2016 年 7 月北京第 1 次印刷
- 著作权合同登记号 图字：01-2015-2962 号
-

定价：69.00 元

读者服务热线：(010) 81055410 印装质量热线：(010) 81055316
反盗版热线：(010) 81055315

目 录

第一部分 Python 编程基础

第 1 章 Python 基础	3	2.2 比较操作符	19
1.1 在交互式环境中输入表达式	3	2.3 布尔操作符	20
1.2 整型、浮点型和字符串数据类型	6	2.3.1 二元布尔操作符	20
1.3 字符串连接和复制	6	2.3.2 not 操作符	21
1.4 在变量中保存值	7	2.4 混合布尔和比较操作符	21
1.4.1 赋值语句	7	2.5 控制流的元素	22
1.4.2 变量名	9	2.5.1 条件	22
1.5 第一个程序	9	2.5.2 代码块	22
1.6 程序剖析	11	2.6 程序执行	23
1.6.1 注释	11	2.7 控制流语句	23
1.6.2 print()函数	11	2.7.1 if 语句	23
1.6.3 input()函数	11	2.7.2 else 语句	24
1.6.4 打印用户的名字	12	2.7.3 elif 语句	25
1.6.5 len()函数	12	2.7.4 while 循环语句	30
1.6.6 str()、int()和 float()函数	13	2.7.5 恼人的循环	31
1.7 小结	15	2.7.6 break 语句	33
1.8 习题	15	2.7.7 continue 语句	34
第 2 章 控制流	17	2.7.8 for 循环和 range()函数	37
2.1 布尔值	18	2.7.9 等价的 while 循环	39

2.7.10	range()的开始、停止和 步长参数	39	4.1.5	用下标改变列表中的值	62
2.8	导入模块	40	4.1.6	列表连接和列表复制	62
	from import 语句	41	4.1.7	用 del 语句从列表中 删除值	63
2.9	用 sys.exit()提前结束程序	41	4.2	使用列表	63
2.10	小结	41	4.2.1	列表用于循环	64
2.11	习题	41	4.2.2	in 和 not in 操作符	65
第3章	函数	43	4.2.3	多重赋值技巧	66
3.1	def 语句和参数	44	4.3	增强的赋值操作	66
3.2	返回值和 return 语句	45	4.4	方法	67
3.3	None 值	46	4.4.1	用 index()方法在列表中 查找值	67
3.4	关键字参数和 print()	47	4.4.2	用 append()和 insert()方法在 列表中添加值	68
3.5	局部和全局作用域	48	4.4.3	用 remove()方法从列表中 删除值	69
3.5.1	局部变量不能在全局作用 域内使用	48	4.4.4	用 sort()方法将列表中的值 排序	69
3.5.2	局部作用域不能使用其他 局部作用域内的变量	49	4.5	例子程序：神奇 8 球和列表	70
3.5.3	全局变量可以在局部作用 域中读取	49	4.6	类似列表的类型：字符串和 元组	71
3.5.4	名称相同的局部变量和全局 变量	50	4.6.1	可变和不可变数据类型	72
3.6	global 语句	50	4.6.2	元组数据类型	73
3.7	异常处理	52	4.6.3	用 list()和 tuple()函数来 转换类型	74
3.8	一个小程序：猜数字	54	4.7	引用	75
3.9	小结	55	4.7.1	传递引用	76
3.10	习题	56	4.7.2	copy 模块的 copy()和 deepcopy()函数	77
3.11	实践项目	56	4.8	小结	78
3.11.1	Collatz 序列	56	4.9	习题	78
3.11.2	输入验证	57	4.10	实践项目	79
第4章	列表	59	4.10.1	逗号代码	79
4.1	列表数据类型	59	4.10.2	字符图网格	79
4.1.1	用下标取得列表中的 单个值	60	第5章	字典和结构化数据	81
4.1.2	负数下标	61	5.1	字典数据类型	81
4.1.3	利用切片取得子列表	61			
4.1.4	用 len()取得列表的长度	62			

5.1.1	字典与列表	82	6.2.1	字符串方法 upper()、lower()、isupper()和 islower()	99
5.1.2	keys()、values()和 items()方法	83	6.2.2	isX 字符串方法	100
5.1.3	检查字典中是否存在键或值	84	6.2.3	字符串方法 startswith()和 endswith()	102
5.1.4	get()方法	84	6.2.4	字符串方法 join()和 split()	102
5.1.5	setdefault()方法	85	6.2.5	用 rjust()、ljust()和 center()方法对齐文本	103
5.2	漂亮打印	86	6.2.6	用 strip()、rstrip()和 lstrip()删除空白字符	104
5.3	使用数据结构对真实世界建模	87	6.2.7	用 pyperclip 模块拷贝粘贴字符串	105
5.3.1	井字棋盘	88	6.3	项目：口令保管箱	106
5.3.2	嵌套的字典和列表	91	第 1 步：程序设计和数据结构	106	
5.4	小结	92	第 2 步：处理命令行参数	106	
5.5	习题	93	第 3 步：复制正确的口令	107	
5.6	实践项目	93	6.4	项目：在 Wiki 标记中添加无序列表	108
5.6.1	好玩游戏的物品清单	93	第 1 步：从剪贴板中复制和粘贴	108	
5.6.2	列表到字典的函数，针对好玩游戏物品清单	94	第 2 步：分离文本中的行，并添加星号	109	
5.6.3			第 3 步：连接修改过的行	109	
第 6 章	字符串操作	95	6.5	小结	110
6.1	处理字符串	95	6.6	习题	110
6.1.1	字符串字面量	95	6.7	实践项目	111
6.1.2	双引号	96	表格打印	111	
6.1.3	转义字符	96			
6.1.4	原始字符串	96			
6.1.5	用三重引号的多行字符串	97			
6.1.6	多行注释	97			
6.1.7	字符串下标和切片	98			
6.1.8	字符串的 in 和 not in 操作符	98			
6.2	有用的字符串方法	99			

第二部分 自动化任务

第 7 章	模式匹配与正则表达式	115	7.2	用正则表达式查找文本模式	117
7.1	不用正则表达式来查找文本模式	116	7.2.1	创建正则表达式对象	118
			7.2.2	匹配 Regex 对象	118
			7.2.3	正则表达式匹配复习	119

7.3	用正则表达式匹配更多模式	119	7.18.1	强口令检测	136
7.3.1	利用括号分组	119	7.18.2	strip()的正则表达式版本	136
7.3.2	用管道匹配多个分组	120			
7.3.3	用问号实现可选匹配	121	第 8 章	读写文件	137
7.3.4	用星号匹配零次或多次	121	8.1	文件与文件路径	137
7.3.5	用加号匹配一次或多次	122	8.1.1	Windows 上的倒斜杠以及 OS X 和 Linux 上的正斜杠	138
7.3.6	用花括号匹配特定次数	122	8.1.2	当前工作目录	139
7.4	贪心和非贪心匹配	123	8.1.3	绝对路径与相对路径	139
7.5	findall()方法	124	8.1.4	用 os.makedirs()创建新文件夹	140
7.6	字符分类	124	8.1.5	os.path 模块	140
7.7	建立自己的字符分类	125	8.1.6	处理绝对路径和相对路径	141
7.8	插入字符和美元字符	126	8.1.7	查看文件大小和文件夹内容	142
7.9	通配字符	126	8.1.8	检查路径有效性	143
7.9.1	用点-星匹配所有字符	127	8.2	文件读写过程	144
7.9.2	用句点字符匹配换行	127	8.2.1	用 open()函数打开文件	145
7.10	正则表达式符号复习	128	8.2.2	读取文件内容	145
7.11	不区分大小写的匹配	128	8.2.3	写入文件	146
7.12	用 sub()方法替换字符串	129	8.3	用 shelve 模块保存变量	147
7.13	管理复杂的正则表达式	129	8.4	用 pprint.pformat()函数保存变量	148
7.14	组合使用 re.IGNORECASE、re.DOTALL 和 re.VERBOSE	130	8.5	项目：生成随机的测验试卷文件	149
7.15	项目：电话号码和 E-mail 地址提取程序	130	第 1 步：	将测验数据保存在一个字典中	149
第 1 步：	为电话号码创建一个正则表达式	131	第 2 步：	创建测验文件，并打乱问题的次序	150
第 2 步：	为 E-mail 地址创建一个正则表达式	132	第 3 步：	创建答案选项	151
第 3 步：	在剪贴板文本中找到所有匹配	132	第 4 步：	将内容写入测验试卷和答案文件	151
第 4 步：	所有匹配连接成一个字符串，复制到剪贴板	133	8.6	项目：多重剪贴板	153
第 5 步：	运行程序	133			
第 6 步：	类似程序的构想	134			
7.16	小结	134			
7.17	习题	134			
7.18	实践项目	136			

第 1 步: 注释和 shelf 设置	153	第 1 步: 弄清楚 ZIP 文件的	
第 2 步: 用一个关键字保存剪贴板		名称	168
内容	154	第 2 步: 创建新 ZIP 文件	169
第 3 步: 列出关键字和加载关键字的		第 3 步: 遍历目录树并添加到	
内容	154	ZIP 文件	170
8.7 小结	155	第 4 步: 类似程序的想法	170
8.8 习题	155	9.6 小结	171
8.9 实践项目	156	9.7 习题	171
8.9.1 扩展多重剪贴板	156	9.8 实践项目	171
8.9.2 疯狂填词	156	9.8.1 选择性拷贝	171
8.9.3 正则表达式查找	156	9.8.2 删除不需要的文件	172
第 9 章 组织文件	157	9.8.3 消除缺失的编号	172
9.1 shutil 模块	158	第 10 章 调试	173
9.1.1 复制文件和文件夹	158	10.1 抛出异常	174
9.1.2 文件和文件夹的移动与		10.2 取得反向跟踪的字符串	175
改名	158	10.3 断言	176
9.1.3 永久删除文件和文件夹	160	10.3.1 在交通灯模拟中使用	
9.1.4 用 send2trash 模块安全地		断言	177
删除	160	10.3.2 禁用断言	178
9.2 遍历目录树	161	10.4 日志	178
9.3 用 zipfile 模块压缩文件	162	10.4.1 使用日志模块	178
9.3.1 读取 ZIP 文件	163	10.4.2 不要用 print() 调试	180
9.3.2 从 ZIP 文件中解压缩	164	10.4.3 日志级别	180
9.3.3 创建和添加到 ZIP 文件	164	10.4.4 禁用日志	181
9.4 项目: 将带有美国风格日期的		10.4.5 将日志记录到文件	182
文件改名为欧洲风格日期	165	10.5 IDLE 的调试器	182
第 1 步: 为美国风格的日期创建一个		10.5.1 Go	183
正则表达式	165	10.5.2 Step	183
第 2 步: 识别文件名中的日期		10.5.3 Over	183
部分	166	10.5.4 Out	183
第 3 步: 构成新文件名, 并对文件		10.5.5 Quit	183
改名	167	10.5.6 调试一个数字相加的	
第 4 步: 类似程序的想法	168	程序	184
9.5 项目: 将一个文件夹备份到一个		10.5.7 断点	185
ZIP 文件	168	10.6 小结	187

10.7	习题	187			
10.8	实践项目	188			
第 11 章	从 Web 抓取信息	189			
11.1	项目：利用 webbrowser 模块的 mapIt.py	190			
	第 1 步：弄清楚 URL	190			
	第 2 步：处理命令行参数	191			
	第 3 步：处理剪贴板内容，加载 浏览器	191			
	第 4 步：类似程序的想法	192			
11.2	用 requests 模块从 Web 下载 文件	192			
11.2.1	用 requests.get()函数下载 一个网页	193			
11.2.2	检查错误	193			
11.3	将下载的文件保存到硬盘	194			
11.4	HTML	195			
11.4.1	学习 HTML 的资源	195			
11.4.2	快速复习	195			
11.4.3	查看网页的 HTML 源代码	196			
11.4.4	打开浏览器的开发者 工具	197			
11.4.5	使用开发者工具来寻找 HTML 元素	198			
11.5	用 BeautifulSoup 模块解析 HTML	199			
11.5.1	从 HTML 创建一个 BeautifulSoup 对象	200			
11.5.2	用 select()方法寻找元素	200			
11.5.3	通过元素的属性获取 数据	202			
11.6	项目：“I’m Feeling Lucky” Google 查找	202			
	第 1 步：获取命令行参数，并请求 查找页面	203			
	第 2 步：找到所有的结果	203			
	第 3 步：针对每个结果打开 Web 浏览器	204			
	第 4 步：类似程序的想法	205			
11.7	项目：下载所有 XKCD 漫画	205			
	第 1 步：设计程序	206			
	第 2 步：下载网页	207			
	第 3 步：寻找和下载漫画图像	207			
	第 4 步：保存图像，找到前一张 漫画	208			
	第 5 步：类似程序的想法	209			
11.8	用 selenium 模块控制浏览器	210			
11.8.1	启动 selenium 控制的 浏览器	210			
11.8.2	在页面中寻找元素	211			
11.8.3	点击页面	212			
11.8.4	填写并提交表单	212			
11.8.5	发送特殊键	213			
11.8.6	点击浏览器按钮	213			
11.8.7	关于 selenium 的更多 信息	214			
11.9	小结	214			
11.10	习题	214			
11.11	实践项目	215			
	11.11.1 命令行邮件程序	215			
	11.11.2 图像网站下载	215			
	11.11.3 2048	215			
	11.11.4 链接验证	215			
第 12 章	处理 Excel 电子表格	217			
12.1	Excel 文档	217			
12.2	安装 openpyxl 模块	218			
12.3	读取 Excel 文档	218			
	12.3.1 用 openpyxl 模块打开 Excel				

文档	219	12.13.2 空行插入程序	241
12.3.2 从工作簿中取得工作表	219	12.13.3 电子表格单元格翻转	
12.3.3 从表中取得单元格	220	程序	242
12.3.4 列字母和数字之间的		12.13.4 文本文件到电子表格	242
转换	221	12.13.5 电子表格到文本文件	242
12.3.5 从表中取得行和列	222	第 13 章 处理 PDF 和 Word 文档	243
12.3.6 工作簿、工作表、		13.1 PDF 文档	243
单元格	223	13.1.1 从 PDF 提取文本	244
12.4 项目：从电子表格中读取		13.1.2 解密 PDF	245
数据	223	13.1.3 创建 PDF	246
第 1 步：读取电子表格数据	224	13.1.4 拷贝页面	246
第 2 步：填充数据结构	225	13.1.5 旋转页面	247
第 3 步：将结果写入文件	226	13.1.6 叠加页面	248
第 4 步：类似程序的思想	227	13.1.7 加密 PDF	249
12.5 写入 Excel 文档	227	13.2 项目：从多个 PDF 中合并	
12.5.1 创建并保存 Excel 文档	227	选择的页面	250
12.5.2 创建和删除工作表	228	第 1 步：找到所有 PDF 文件	250
12.5.3 将值写入单元格	229	第 2 步：打开每个 PDF 文件	251
12.6 项目：更新一个电子表格	229	第 3 步：添加每一页	252
第 1 步：利用更新信息建立数据		第 4 步：保存结果	252
结构	230	第 5 步：类似程序的想法	253
第 2 步：检查所有行，更新不正确的		13.3 Word 文档	253
价格	231	13.3.1 读取 Word 文档	254
第 3 步：类似程序的思想	231	13.3.2 从.docx 文件中取得完整的	
12.7 设置单元格的字体风格	232	文本	254
12.8 Font 对象	232	13.3.3 设置 Paragraph 和 Run 对象	
12.9 公式	234	的样式	255
12.10 调整行和列	235	13.3.4 创建带有非默认样式的	
12.10.1 设置行高和列宽	235	Word 文档	257
12.10.2 合并和拆分单元格	236	13.3.5 Run 属性	257
12.10.3 冻结窗格	237	13.3.6 写入 Word 文档	258
12.10.4 图表	238	13.3.7 添加标题	260
12.11 小结	240	13.3.8 添加换行符和换页符	261
12.12 习题	240	13.3.9 添加图像	261
12.13 实践项目	241	13.4 小结	262
12.13.1 乘法表	241		

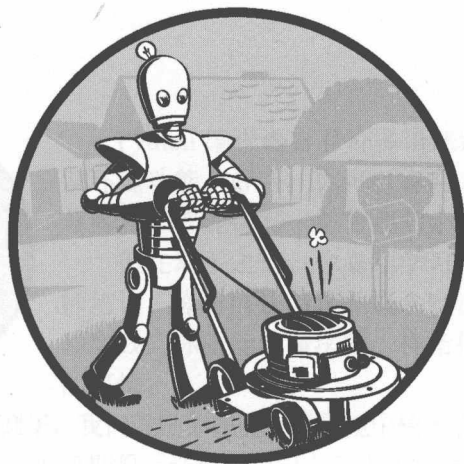
13.5 习题	262	第 4 步: 类似程序的想法	277
13.6 实践项目	263	14.6 小结	277
13.6.1 PDF 偏执狂	263	14.7 习题	277
13.6.2 定制邀请函, 保存为 Word 文档	263	14.8 实践项目	277
13.6.3 暴力 PDF 口令破解程序	264	第 15 章 保持时间、计划任务和启动程序	279
第 14 章 处理 CSV 文件和 JSON 数据	265	15.1 time 模块	279
14.1 csv 模块	265	15.1.1 time.time()函数	279
14.1.1 Reader 对象	266	15.1.2 time.sleep()函数	280
14.1.2 在 for 循环中, 从 Reader 对象读取数据	267	15.2 数字四舍五入	281
14.1.3 Writer 对象	268	15.3 项目: 超级秒表	282
14.1.4 delimiter 和 lineterminator 关键字参数	269	第 1 步: 设置程序来记录时间	282
14.2 项目: 从 CSV 文件中删除表头	269	第 2 步: 记录并打印单圈时间	283
第 1 步: 循环遍历每个 CSV 文件	270	第 3 步: 类似程序的想法	283
第 2 步: 读入 CSV 文件	270	15.4 datetime 模块	284
第 3 步: 写入 CSV 文件, 没有第一行	271	15.4.1 timedelta 数据类型	285
第 4 步: 类似程序的想法	272	15.4.2 暂停直至特定日期	286
14.3 JSON 和 API	272	15.4.3 将 datetime 对象转换为字符串	287
14.4 json 模块	273	15.4.4 将字符串转换成 datetime 对象	288
14.4.1 用 loads()函数读取 JSON	273	15.5 回顾 Python 的时间函数	288
14.4.2 用 dumps 函数写出 JSON	273	15.6 多线程	289
14.5 项目: 取得当前的天气数据	274	15.6.1 向线程的目标函数传递参数	290
第 1 步: 从命令行参数获取位置	274	15.6.2 并发问题	291
第 2 步: 下载 JSON 数据	275	15.7 项目: 多线程 XKCD 下载程序	291
第 3 步: 加载 JSON 数据并打印天气	275	第 1 步: 修改程序以使用函数	292
		第 2 步: 创建并启动线程	293
		第 3 步: 等待所有线程结束	293
		15.8 从 Python 启动其他程序	294
		15.8.1 向 Popen()传递命令行参数	295
		15.8.2 Task Scheduler、launchd 和	

cron	296	邮件地址	313
15.8.3 用 Python 打开网站	296	16.4.9 从原始消息中获取正文	314
15.8.4 运行其他 Python 脚本	296	16.4.10 删除电子邮件	315
15.8.5 用默认的应用程序打开		16.4.11 从 IMAP 服务器断开	315
文件	297	16.5 项目：向会员发送会费提醒	
15.9 项目：简单的倒计时程序	298	电子邮件	316
第 1 步：倒计时	298	第 1 步：打开 Excel 文件	316
第 2 步：播放声音文件	298	第 2 步：查找所有未付成员	317
第 3 步：类似程序的想法	299	第 3 步：发送定制的电子邮	
15.10 小结	299	件提醒	318
15.11 习题	300	16.6 用 Twilio 发送短信	319
15.12 实践项目	300	16.6.1 注册 Twilio 账号	319
15.12.1 美化的秒表	300	16.6.2 发送短信	320
15.12.2 计划的 Web 漫画下载	301	16.7 项目：“只给我发短信”	
第 16 章 发送电子邮件和短信	303	模块	321
16.1 SMTP	303	16.8 小结	322
16.2 发送电子邮件	304	16.9 习题	323
16.2.1 连接到 SMTP 服务器	304	16.10 实践项目	323
16.2.2 发送 SMTP 的“Hello”		16.10.1 随机分配家务活的电子	
消息	305	邮件程序	323
16.2.3 开始 TLS 加密	306	16.10.2 伞提醒程序	324
16.2.4 登录到 SMTP 服务器	306	16.10.3 自动退订	324
16.2.5 发送电子邮件	306	16.10.4 通过电子邮件控制你的	
16.2.6 从 SMTP 服务器断开	307	电脑	324
16.3 IMAP	307	第 17 章 操作图像	327
16.4 用 IMAP 获取和删除电子		17.1 计算机图像基础	327
邮件	307	17.1.1 颜色和 RGBA 值	328
16.4.1 连接到 IMAP 服务器	308	17.1.2 坐标和 Box 元组	329
16.4.2 登录到 IMAP 服务器	309	17.2 用 Pillow 操作图像	330
16.4.3 搜索电子邮件	309	17.2.1 处理 Image 数据类型	331
16.4.4 选择文件夹	309	17.2.2 裁剪图片	332
16.4.5 执行搜索	310	17.2.3 复制和粘贴图像到其他	
16.4.6 大小限制	312	图像	333
16.4.7 取邮件并标记为已读	312	17.2.4 调整图像大小	335
16.4.8 从原始消息中获取电子		17.2.5 旋转和翻转图像	336

17.2.6 更改单个像素	338	第 3 步: 获取并打印鼠标坐标	356
17.3 项目: 添加徽标	339	18.5 控制鼠标交互	357
第 1 步: 打开徽标图像	340	18.5.1 点击鼠标	357
第 2 步: 遍历所有文件并打开 图像	341	18.5.2 拖动鼠标	357
第 3 步: 调整图像的大小	341	18.5.3 滚动鼠标	359
第 4 步: 添加徽标, 并保存 更改	342	18.6 处理屏幕	360
第 5 步: 类似程序的想法	343	18.6.1 获取屏幕快照	360
17.4 在图像上绘画	344	18.6.2 分析屏幕快照	360
17.4.1 绘制形状	344	18.7 项目: 扩展 mouseNow 程序	361
17.4.2 绘制文本	346	18.8 图像识别	362
17.5 小结	347	18.9 控制键盘	363
17.6 习题	348	18.9.1 通过键盘发送一个 字符串	363
17.7 实践项目	348	18.9.2 键名	364
17.7.1 扩展和修正本章项目的 程序	348	18.9.3 按下和释放键盘	365
17.7.2 在硬盘上识别照片 文件夹	349	18.9.4 热键组合	365
17.7.3 定制的座位卡	350	18.10 复习 PyAutoGUI 的函数	366
第 18 章 用 GUI 自动化控制键盘和 鼠标	351	18.11 项目: 自动填表程序	367
18.1 安装 pyautogui 模块	351	第 1 步: 弄清楚步骤	368
18.2 走对路	352	第 2 步: 建立坐标	368
18.2.1 通过注销关闭所有程序	352	第 3 步: 开始键入数据	370
18.2.2 暂停和自动防故障装置	352	第 4 步: 处理选择列表和单选 按钮	371
18.3 控制鼠标移动	353	第 5 步: 提交表单并等待	372
18.3.1 移动鼠标	354	18.12 小结	372
18.3.2 获取鼠标位置	354	18.13 习题	373
18.4 项目: “现在鼠标在 哪里?”	355	18.14 实践项目	373
第 1 步: 导入模块	355	18.14.1 看起来很忙	373
第 2 步: 编写退出代码和无限 循环	355	18.14.2 即时通信机器人	373
		18.14.3 玩游戏机器人指南	374
		附录 A 安装第三方模块	375
		附录 B 运行程序	377
		附录 C 习题答案	381

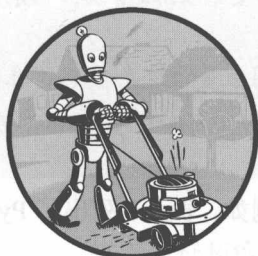
第一部分

Python 编程基础



第 1 章

Python 基础



Python 编程语言有许多语法结构、标准库函数和交互式开发环境功能。好在，你可以忽略大多数内容。你只需要学习部分内容，就能编写一些方便的小程序。

但在动手之前，你必须学习一些基本编程概念。就像魔法师培训，你可能认为这些概念既深奥又啰嗦，但有了一些知识和实践，你就能像魔法师一样指挥你的计算机，完成难以置信的事情。

本章有几个例子，我们鼓励你在交互式环境中输入它们。交互式环境让你每次执行一条 Python 指令，并立即显示结果。使用交互式环境对于了解基本 Python 指令的行为是很好的，所以你在阅读时要试一下。做过的事比仅仅读过的内容，更令人印象深刻。

1.1 在交互式环境中输入表达式

启动 IDLE 就运行了交互式环境，这是和 Python 一起安装的。在 Windows 上，打开“开始”菜单，选择“All Programs ▶ Python 3.3”，然后选择“IDLE (Python GUI)”。在 OS X 上，选择“Applications ▶ MacPython 3.3 ▶ IDLE”。在 Ubuntu 上，打开新的终端窗口并输入 `idle3`。

一个窗口会出现，包含>>>提示符，这就是交互式环境。在提示符后输入 2 + 2，让 Python 做一些简单的算术。

```
>>> 2 + 2
4
```

IDLE 窗口现在应该显示下面这样的文本：

```
Python 3.3.2 (v3.3.2:d047928ae3f6, May 16 2013, 00:06:53) [MSC v.1600 64 bit
(AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> 2 + 2
4
>>>
```

在 Python 中，2 + 2 称为“表达式”，它是语言中最基本的编程结构。表达式包含“值”（例如 2）和“操作符”（例如+），并且总是可以求值（也就是归约）为单个值。这意味着在 Python 代码中，所有使用表达式的地方，也可以使用一个值。

在前面的例子中，2 + 2 被求值为单个值 4。没有操作符的单个值也被认为是一个表达式，尽管它求值的结果就是它自己，像下面这样：

```
>>> 2
2
```

错误没关系！

如果程序包含计算机不能理解的代码，就会崩溃，这将导致 Python 显示错误信息。错误信息并不会破坏你的计算机，所以不要害怕犯错误。“崩溃”只是意味着程序意外地停止执行。

如果你希望对一条错误信息了解更多，可以在网上查找这条信息的准确文本，找到关于这个错误的更多内容。也可以查看 <http://nostarch.com/automatestuff/>，这里有常见的 Python 错误信息和含义的列表。

Python 表达式中也可以使用大量其他操作符。例如，表 1-1 列出了 Python 的所有数学操作符。

表 1-1 数学操作符，优先级从高到低

操作符	操作	例子	求值为
**	指数	2 ** 3	8
%	取模/取余数	22 % 8	6
//	整除/商数取整	22 // 8	2
/	除法	22 / 8	2.75
*	乘法	3 * 5	15
-	减法	5 - 2	3
+	加法	2 + 2	4

数学操作符的操作顺序（也称为“优先级”）与数学中类似。**操作符首先求值，接下来是*、/、//和%操作符，从左到右。+和-操作符最后求值，也是从左到右。