

Python编程快速上手——让繁琐工作自动化

作者：斯维加特(Al Sweigart)

目 录

- [版权信息](#)
- [版权声明](#)
- [内容提要](#)
- [作者简介](#)
- [技术评审者简介](#)
- [致谢](#)
- [译者序 会编程的人不一样](#)
- [前言](#)
- [本书的读者对象](#)
- [编码规范](#)
- [什么是编程](#)
- [什么是Python](#)
- [程序员不需要知道太多数学](#)
- [编程是创造性活动](#)
- [本书简介](#)
- [下载和安装Python](#)
- [启动IDLE](#)
- [交互式环境](#)
- [如何寻求帮助](#)
- [聪明地提出编程问题](#)
- [小结](#)
- [第一部分 Python编程基础](#)
- [第1章 Python基础](#)
 - [1.1 在交互式环境中输入表达式](#)
 - [1.2 整型、浮点型和字符串数据类型](#)
 - [1.3 字符串连接和复制](#)
 - [1.4 在变量中保存值](#)
 - [1.4.1 赋值语句](#)
 - [1.4.2 变量名](#)
 - [1.5 第一个程序](#)
 - [1.6 程序剖析](#)
 - [1.6.1 注释](#)
 - [1.6.2 print\(\)函数](#)
 - [1.6.3 input\(\)函数](#)
 - [1.6.4 打印用户的名字](#)
 - [1.6.5 len\(\)函数](#)
 - [1.6.6 str\(\)、int\(\)和float\(\)函数](#)
 - [1.7 小结](#)
 - [1.8 习题](#)
- [第2章 控制流](#)
 - [2.1 布尔值](#)
 - [2.2 比较操作符](#)
 - [2.3 布尔操作符](#)
 - [2.3.1 二元布尔操作符](#)
 - [2.3.2 not操作符](#)
 - [2.4 混合布尔和比较操作符](#)
 - [2.5 控制流的元素](#)
 - [2.5.1 条件](#)
 - [2.5.2 代码块](#)
 - [2.6 程序执行](#)
 - [2.7 控制流语句](#)
 - [2.7.1 if语句](#)
 - [2.7.2 else语句](#)
 - [2.7.3 elif语句](#)

- [2.7.4 while循环语句](#)
- [2.7.5 恼人的循环](#)
- [2.7.6 break语句](#)
- [2.7.7 continue语句](#)
- [2.7.8 for循环和range\(\)函数](#)
- [2.7.9 等价的while循环](#)
- [2.7.10 range\(\)的开始、停止和步长参数](#)
- [2.8 导入模块](#)
- [from import语句](#)
- [2.9 用sys.exit\(\)提前结束程序](#)
- [2.10 小结](#)
- [2.11 习题](#)
- [第3章 函数](#)
- [3.1 def语句和参数](#)
- [3.2 返回值和return语句](#)
- [3.3 None值](#)
- [3.4 关键字参数和print\(\)](#)
- [3.5 局部和全局作用域](#)
 - [3.5.1 局部变量不能在全局作用域内使用](#)
 - [3.5.2 局部作用域不能使用其他局部作用域内的变量](#)
 - [3.5.3 全局变量可以在局部作用域中读取](#)
 - [3.5.4 名称相同的局部变量和全局变量](#)
- [3.6 global语句](#)
- [3.7 异常处理](#)
- [3.8 一个小程序：猜数字](#)
- [3.9 小结](#)
- [3.10 习题](#)
- [3.11 实践项目](#)
 - [3.11.1 Collatz序列](#)
 - [3.11.2 输入验证](#)
- [第4章 列表](#)
- [4.1 列表数据类型](#)
 - [4.1.1 用下标取得列表中的单个值](#)
 - [4.1.2 负数下标](#)
 - [4.1.3 利用切片取得子列表](#)
 - [4.1.4 用len\(\)取得列表的长度](#)
 - [4.1.5 用下标改变列表中的值](#)
 - [4.1.6 列表连接和列表复制](#)
 - [4.1.7 用del语句从列表中删除值](#)
- [4.2 使用列表](#)
 - [4.2.1 列表用于循环](#)
 - [4.2.2 in和not in操作符](#)
 - [4.2.3 多重赋值技巧](#)
- [4.3 增强的赋值操作](#)
- [4.4 方法](#)
 - [4.4.1 用index\(\)方法在列表中查找值](#)
 - [4.4.2 用append\(\)和insert\(\)方法在列表中添加值](#)
 - [4.4.3 用remove\(\)方法从列表中删除值](#)
 - [4.4.4 用sort\(\)方法将列表中的值排序](#)
- [4.5 例子程序：神奇8球和列表](#)
- [4.6 类似列表的类型：字符串和元组](#)
 - [4.6.1 可变和不可变数据类型](#)
 - [4.6.2 元组数据类型](#)
 - [4.6.3 用list\(\)和tuple\(\)函数来转换类型](#)
- [4.7 引用](#)
 - [4.7.1 传递引用](#)
 - [4.7.2 copy模块的copy\(\)和deepcopy\(\)函数](#)
- [4.8 小结](#)
- [4.9 习题](#)
- [4.10 实践项目](#)

- [4.10.1 逗号代码](#)
- [4.10.2 字符图网格](#)
- [第5章 字典和结构化数据](#)
- [5.1 字典数据类型](#)
- [5.1.1 字典与列表](#)
- [5.1.2 keys\(\)、values\(\)和items\(\)方法](#)
- [5.1.3 检查字典中是否存在键或值](#)
- [5.1.4 get\(\)方法](#)
- [5.1.5.setdefault\(\)方法](#)
- [5.2 漂亮打印](#)
- [5.3 使用数据结构对真实世界建模](#)
- [5.3.1 井字棋盘](#)
- [5.3.2 嵌套的字典和列表](#)
- [5.4 小结](#)
- [5.5 习题](#)
- [5.6 实践项目](#)
- [5.6.1 好玩游戏的物品清单](#)
- [5.6.2 列表到字典的函数，针对好玩游戏物品清单](#)
- [第6章 字符串操作](#)
- [6.1 处理字符串](#)
- [6.1.1 字符串字面量](#)
- [6.1.2 双引号](#)
- [6.1.3 转义字符](#)
- [6.1.4 原始字符串](#)
- [6.1.5 用三重引号的多行字符串](#)
- [6.1.6 多行注释](#)
- [6.1.7 字符串下标和切片](#)
- [6.1.8 字符串的in和not in操作符](#)
- [6.2 有用的字符串方法](#)
- [6.2.1 字符串方法upper\(\)、lower\(\)、isupper\(\)和islower\(\)](#)
- [6.2.2 isX字符串方法](#)
- [6.2.3 字符串方法startswith\(\)和endswith\(\)](#)
- [6.2.4 字符串方法join\(\)和split\(\)](#)
- [6.2.5 用rjust\(\)、ljust\(\)和center\(\)方法对齐文本](#)
- [6.2.6 用strip\(\)、rstrip\(\)和lstrip\(\)删除空白字符](#)
- [6.2.7 用pyperclip模块拷贝粘贴字符串](#)
- [6.3 项目：口令保管箱](#)
- [第1步：程序设计和数据结构](#)
- [第2步：处理命令行参数](#)
- [第3步：复制正确的口令](#)
- [6.4 项目：在Wiki标记中添加无序列表](#)
- [第1步：从剪贴板中复制和粘贴](#)
- [第2步：分离文本中的行，并添加星号](#)
- [第3步：连接修改过的行](#)
- [6.5 小结](#)
- [6.6 习题](#)
- [6.7 实践项目](#)
- [表格打印](#)
- [第二部分 自动化任务](#)
- [第7章 模式匹配与正则表达式](#)
- [7.1 不用正则表达式来查找文本模式](#)
- [7.2 用正则表达式查找文本模式](#)
- [7.2.1 创建正则表达式对象](#)
- [7.2.2 匹配Regex对象](#)
- [7.2.3 正则表达式匹配复习](#)
- [7.3 用正则表达式匹配更多模式](#)
- [7.3.1 利用括号分组](#)
- [7.3.2 用管道匹配多个分组](#)
- [7.3.3 用问号实现可选匹配](#)
- [7.3.4 用星号匹配零次或多次](#)

- [7.3.5 用加号匹配一次或多次](#)
- [7.3.6 用花括号匹配特定次数](#)
- [7.4 贪心和非贪心匹配](#)
- [7.5 findall\(\)方法](#)
- [7.6 字符分类](#)
- [7.7 建立自己的字符分类](#)
- [7.8 插入字符和美元字符](#)
- [7.9 通配字符](#)
 - [7.9.1 用点-星匹配所有字符](#)
 - [7.9.2 用句点字符匹配换行](#)
- [7.10 正则表达式符号复习](#)
- [7.11 不区分大小写的匹配](#)
- [7.12 用sub\(\)方法替换字符串](#)
- [7.13 管理复杂的正则表达式](#)
- [7.14 组合使用re.IGNORECASE、re.DOTALL和re.VERBOSE](#)
- [7.15 项目：电话号码和E-mail地址提取程序](#)
 - [第1步：为电话号码创建一个正则表达式](#)
 - [第2步：为E-mail地址创建一个正则表达式](#)
 - [第3步：在剪贴板文本中找到所有匹配](#)
 - [第4步：所有匹配连接成一个字符串，复制到剪贴板](#)
 - [第5步：运行程序](#)
 - [第6步：类似程序的构想](#)
- [7.16 小结](#)
- [7.17 习题](#)
- [7.18 实践项目](#)
 - [7.18.1 强口令检测](#)
 - [7.18.2 strip\(\)的正则表达式版本](#)

[第8章 读写文件](#)

- [8.1 文件与文件路径](#)
 - [8.1.1 Windows上的倒斜杠以及OS X和Linux上的正斜杠](#)
 - [8.1.2 当前工作目录](#)
 - [8.1.3 绝对路径与相对路径](#)
 - [8.1.4 用os.makedirs\(\)创建新文件夹](#)
 - [8.1.5 os.path模块](#)
 - [8.1.6 处理绝对路径和相对路径](#)
 - [8.1.7 查看文件大小和文件夹内容](#)
 - [8.1.8 检查路径有效性](#)
- [8.2 文件读写过程](#)
 - [8.2.1 用open\(\)函数打开文件](#)
 - [8.2.2 读取文件内容](#)
 - [8.2.3 写入文件](#)
- [8.3 用shelve模块保存变量](#)
- [8.4 用pprint.pformat\(\)函数保存变量](#)
- [8.5 项目：生成随机的测验试卷文件](#)
 - [第1步：将测验数据保存在一个字典中](#)
 - [第2步：创建测验文件，并打乱问题的次序](#)
 - [第3步：创建答案选项](#)
 - [第4步：将内容写入测验试卷和答案文件](#)
- [8.6 项目：多重剪贴板](#)
 - [第1步：注释和shelf设置](#)
 - [第2步：用一个关键字保存剪贴板内容](#)
 - [第3步：列出关键字和加载关键字的内容](#)
- [8.7 小结](#)
- [8.8 习题](#)
- [8.9 实践项目](#)
 - [8.9.1 扩展多重剪贴板](#)
 - [8.9.2 疯狂填词](#)
 - [8.9.3 正则表达式查找](#)

[第9章 组织文件](#)

- [9.1 shutil模块](#)

- [9.1.1 复制文件和文件夹](#)
- [9.1.2 文件和文件夹的移动与改名](#)
- [9.1.3 永久删除文件和文件夹](#)
- [9.1.4 用send2trash模块安全地删除](#)
- [9.2 遍历目录树](#)
- [9.3 用zipfile模块压缩文件](#)
 - [9.3.1 读取ZIP文件](#)
 - [9.3.2 从ZIP文件中解压缩](#)
 - [9.3.3 创建和添加到ZIP文件](#)
- [9.4 项目：将带有美国风格日期的文件改名为欧洲风格日期](#)
 - [第1步：为美国风格的日期创建一个正则表达式](#)
 - [第2步：识别文件名中的日期部分](#)
 - [第3步：构成新文件名，并对文件改名](#)
 - [第4步：类似程序的想法](#)
- [9.5 项目：将一个文件夹备份到一个ZIP文件](#)
 - [第1步：弄清楚ZIP文件的名称](#)
 - [第2步：创建新ZIP文件](#)
 - [第3步：遍历目录树并添加到ZIP文件](#)
 - [第4步：类似程序的想法](#)
- [9.6 小结](#)
- [9.7 习题](#)
- [9.8 实践项目](#)
 - [9.8.1 选择性拷贝](#)
 - [9.8.2 删除不需要的文件](#)
 - [9.8.3 消除缺失的编号](#)
- [第10章 调试](#)
 - [10.1 抛出异常](#)
 - [10.2 取得反向跟踪的字符串](#)
 - [10.3 断言](#)
 - [10.3.1 在交通灯模拟中使用断言](#)
 - [10.3.2 禁用断言](#)
 - [10.4 日志](#)
 - [10.4.1 使用日志模块](#)
 - [10.4.2 不要用print\(\)调试](#)
 - [10.4.3 日志级别](#)
 - [10.4.4 禁用日志](#)
 - [10.4.5 将日志记录到文件](#)
 - [10.5 IDLE的调试器](#)
 - [10.5.1 Go](#)
 - [10.5.2 Step](#)
 - [10.5.3 Over](#)
 - [10.5.4 Out](#)
 - [10.5.5 Quit](#)
 - [10.5.6 调试一个数字相加的程序](#)
 - [10.5.7 断点](#)
 - [10.6 小结](#)
 - [10.7 习题](#)
 - [10.8 实践项目](#)
- [调试硬币抛掷](#)
- [第11章 从Web抓取信息](#)
 - [11.1 项目：利用webbrowser模块的mapIt.py](#)
 - [第1步：弄清楚URL](#)
 - [第2步：处理命令行参数](#)
 - [第3步：处理剪贴板内容，加载浏览器](#)
 - [第4步：类似程序的想法](#)
 - [11.2 用requests模块从Web下载文件](#)
 - [11.2.1 用requests.get\(\)函数下载一个网页](#)
 - [11.2.2 检查错误](#)
 - [11.3 将下载的文件保存到硬盘](#)
 - [11.4 HTML](#)

- [11.4.1 学习HTML的资源](#)
- [11.4.2 快速复习](#)
- [11.4.3 查看网页的HTML源代码](#)
- [11.4.4 打开浏览器的开发者工具](#)
- [11.4.5 使用开发者工具来寻找HTML元素](#)
- [11.5 用BeautifulSoup模块解析HTML](#)
 - [11.5.1 从HTML创建一个BeautifulSoup对象](#)
 - [11.5.2 用select\(\)方法寻找元素](#)
 - [11.5.3 通过元素的属性获取数据](#)
- [11.6 项目：‘I’m Feeling Lucky’Google查找](#)
 - [第1步：获取命令行参数，并请求查找页面](#)
 - [第2步：找到所有的结果](#)
 - [第3步：针对每个结果打开Web浏览器](#)
 - [第4步：类似程序的想法](#)
- [11.7 项目：下载所有XKCD漫画](#)
 - [第1步：设计程序](#)
 - [第2步：下载网页](#)
 - [第3步：寻找和下载漫画图像](#)
 - [第4步：保存图像，找到前一张漫画](#)
 - [第5步：类似程序的想法](#)
- [11.8 用selenium模块控制浏览器](#)
 - [11.8.1 启动selenium控制的浏览器](#)
 - [11.8.2 在页面中寻找元素](#)
 - [11.8.3 点击页面](#)
 - [11.8.4 填写并提交表单](#)
 - [11.8.5 发送特殊键](#)
 - [11.8.6 点击浏览器按钮](#)
 - [11.8.7 关于selenium的更多信息](#)
- [11.9 小结](#)
- [11.10 习题](#)
- [11.11 实践项目](#)
 - [11.11.1 命令行邮件程序](#)
 - [11.11.2 图像网站下载](#)
 - [11.11.3 2048](#)
 - [11.11.4 链接验证](#)
- [第12章 处理Excel电子表格](#)
 - [12.1 Excel文档](#)
 - [12.2 安装openpyxl模块](#)
 - [12.3 读取Excel文档](#)
 - [12.3.1 用openpyxl模块打开Excel文档](#)
 - [12.3.2 从工作簿中取得工作表](#)
 - [12.3.3 从表中取得单元格](#)
 - [12.3.4 列字母和数字之间的转换](#)
 - [12.3.5 从表中取得行和列](#)
 - [12.3.6 工作簿、工作表、单元格](#)
 - [12.4 项目：从电子表格中读取数据](#)
 - [第1步：读取电子表格数据](#)
 - [第2步：填充数据结构](#)
 - [第3步：将结果写入文件](#)
 - [第4步：类似程序的思想](#)
 - [12.5 写入Excel文档](#)
 - [12.5.1 创建并保存Excel文档](#)
 - [12.5.2 创建和删除工作表](#)
 - [12.5.3 将值写入单元格](#)
 - [12.6 项目：更新一个电子表格](#)
 - [第1步：利用更新信息建立数据结构](#)
 - [第2步：检查所有行，更新不正确的价格](#)
 - [第3步：类似程序的思想](#)
 - [12.7 设置单元格的字体风格](#)
 - [12.8 Font对象](#)

- [12.9 公式](#)
- [12.10 调整行和列](#)
 - [12.10.1 设置行高和列宽](#)
 - [12.10.2 合并和拆分单元格](#)
 - [12.10.3 冻结窗格](#)
 - [12.10.4 图表](#)
- [12.11 小结](#)
- [12.12 习题](#)
- [12.13 实践项目](#)
 - [12.13.1 乘法表](#)
 - [12.13.2 空行插入程序](#)
 - [12.13.3 电子表格单元格翻转程序](#)
 - [12.13.4 文本文件到电子表格](#)
 - [12.13.5 电子表格到文本文件](#)
- [第13章 处理PDF和Word文档](#)
 - [13.1 PDF文档](#)
 - [13.1.1 从PDF提取文本](#)
 - [13.1.2 解密PDF](#)
 - [13.1.3 创建PDF](#)
 - [13.1.4 拷贝页面](#)
 - [13.1.5 旋转页面](#)
 - [13.1.6 叠加页面](#)
 - [13.1.7 加密PDF](#)
 - [13.2 项目：从多个PDF中合并选择的页面](#)
 - [第1步：找到所有PDF文件](#)
 - [第2步：打开每个PDF文件](#)
 - [第3步：添加每一页](#)
 - [第4步：保存结果](#)
 - [第5步：类似程序的想法](#)
 - [13.3 Word文档](#)
 - [13.3.1 读取Word文档](#)
 - [13.3.2 从.docx文件中取得完整的文本](#)
 - [13.3.3 设置Paragraph和Run对象的样式](#)
 - [13.3.4 创建带有非默认样式的Word文档](#)
 - [13.3.5 Run属性](#)
 - [13.3.6 写入Word文档](#)
 - [13.3.7 添加标题](#)
 - [13.3.8 添加换行符和换页符](#)
 - [13.3.9 添加图像](#)
 - [13.4 小结](#)
 - [13.5 习题](#)
 - [13.6 实践项目](#)
 - [13.6.1 PDF偏执狂](#)
 - [13.6.2 定制邀请函，保存为Word文档](#)
 - [13.6.3 暴力PDF口令破解程序](#)
- [第14章 处理CSV文件和JSON数据](#)
 - [14.1 csv模块](#)
 - [14.1.1 Reader对象](#)
 - [14.1.2 在for循环中，从Reader对象读取数据](#)
 - [14.1.3 Writer对象](#)
 - [14.1.4 delimiter和lineterminator关键字参数](#)
 - [14.2 项目：从CSV文件中删除表头](#)
 - [第1步：循环遍历每个CSV文件](#)
 - [第2步：读入CSV文件](#)
 - [第3步：写入CSV文件，没有第一行](#)
 - [第4步：类似程序的想法](#)
 - [14.3 JSON和API](#)
 - [14.4 json模块](#)
 - [14.4.1 用loads\(\)函数读取JSON](#)
 - [14.4.2 用dumps函数写出JSON](#)

- [14.5 项目：取得当前的天气数据](#)
- [第1步：从命令行参数获取位置](#)
- [第2步：下载JSON数据](#)
- [第3步：加载JSON数据并打印天气](#)
- [第4步：类似程序的想法](#)
- [14.6 小结](#)
- [14.7 习题](#)
- [14.8 实践项目](#)
- [Excel到CSV的转换程序](#)
- [第15章 保持时间、计划任务和启动程序](#)
- [15.1 time模块](#)
- [15.1.1 time.time\(\)函数](#)
- [15.1.2 time.sleep\(\)函数](#)
- [15.2 数字四舍五入](#)
- [15.3 项目：超级秒表](#)
- [第1步：设置程序来记录时间](#)
- [第2步：记录并打印单圈时间](#)
- [第3步：类似程序的想法](#)
- [15.4 datetime模块](#)
- [15.4.1 timedelta数据类型](#)
- [15.4.2 暂停直至特定日期](#)
- [15.4.3 将datetime对象转换为字符串](#)
- [15.4.4 将字符串转换成datetime对象](#)
- [15.5 回顾Python的时间函数](#)
- [15.6 多线程](#)
- [15.6.1 向线程的目标函数传递参数](#)
- [15.6.2 并发问题](#)
- [15.7 项目：多线程XKCD下载程序](#)
- [第1步：修改程序以使用函数](#)
- [第2步：创建并启动线程](#)
- [第3步：等待所有线程结束](#)
- [15.8 从Python启动其他程序](#)
- [15.8.1 向Popen\(\)传递命令行参数](#)
- [15.8.2 Task Scheduler、launchd和cron](#)
- [15.8.3 用Python打开网站](#)
- [15.8.4 运行其他Python脚本](#)
- [15.8.5 用默认的应用程序打开文件](#)
- [15.9 项目：简单的倒计时程序](#)
- [第1步：倒计时](#)
- [第2步：播放声音文件](#)
- [第3步：类似程序的想法](#)
- [15.10 小结](#)
- [15.11 习题](#)
- [15.12 实践项目](#)
- [15.12.1 美化的秒表](#)
- [15.12.2 计划的Web漫画下载](#)
- [第16章 发送电子邮件和短信](#)
- [16.1 SMTP](#)
- [16.2 发送电子邮件](#)
- [16.2.1 连接到SMTP服务器](#)
- [16.2.2 发送SMTP的“Hello”消息](#)
- [16.2.3 开始TLS加密](#)
- [16.2.4 登录到SMTP服务器](#)
- [16.2.5 发送电子邮件](#)
- [16.2.6 从SMTP服务器断开](#)
- [16.3 IMAP](#)
- [16.4 用IMAP获取和删除电子邮件](#)
- [16.4.1 连接到IMAP服务器](#)
- [16.4.2 登录到IMAP服务器](#)
- [16.4.3 搜索电子邮件](#)

- [16.4.4 选择文件夹](#)
- [16.4.5 执行搜索](#)
- [16.4.6 大小限制](#)
- [16.4.7 取邮件并标记为已读](#)
- [16.4.8 从原始消息中获取电子邮件地址](#)
- [16.4.9 从原始消息中获取正文](#)
- [16.4.10 删除电子邮件](#)
- [16.4.11 从IMAP服务器断开](#)
- [16.5 项目：向会员发送会费提醒电子邮件](#)
- [第1步：打开Excel文件](#)
- [第2步：查找所有未付成员](#)
- [第3步：发送定制的电子邮件提醒](#)
- [16.6 用Twilio发送短信](#)
- [16.6.1 注册Twilio账号](#)
- [16.6.2 发送短信](#)
- [16.7 项目：“只给我发短信”模块](#)
- [16.8 小结](#)
- [16.9 习题](#)
- [16.10 实践项目](#)
- [16.10.1 随机分配家务活的电子邮件程序](#)
- [16.10.2 伞提醒程序](#)
- [16.10.3 自动退订](#)
- [16.10.4 通过电子邮件控制你的电脑](#)
- [第17章 操作图像](#)
- [17.1 计算机图像基础](#)
- [17.1.1 颜色和RGBA值](#)
- [17.1.2 坐标和Box元组](#)
- [17.2 用Pillow操作图像](#)
- [17.2.1 处理Image数据类型](#)
- [17.2.2 裁剪图片](#)
- [17.2.3 复制和粘贴图像到其他图像](#)
- [17.2.4 调整图像大小](#)
- [17.2.5 旋转和翻转图像](#)
- [17.2.6 更改单个像素](#)
- [17.3 项目：添加徽标](#)
- [第1步：打开徽标图像](#)
- [第2步：遍历所有文件并打开图像](#)
- [第3步：调整图像的大小](#)
- [第4步：添加徽标，并保存更改](#)
- [第5步：类似程序的想法](#)
- [17.4 在图像上绘画](#)
- [17.4.1 绘制形状](#)
- [17.4.2 绘制文本](#)
- [17.5 小结](#)
- [17.6 习题](#)
- [17.7 实践项目](#)
- [17.7.1 扩展和修正本章项目的程序](#)
- [17.7.2 在硬盘上识别照片文件夹](#)
- [17.7.3 定制的座位卡](#)
- [第18章 用GUI自动化控制键盘和鼠标](#)
- [18.1 安装pyautogui模块](#)
- [18.2 走对路](#)
- [18.2.1 通过注销关闭所有程序](#)
- [18.2.2 暂停和自动防故障装置](#)
- [18.3 控制鼠标移动](#)
- [18.3.1 移动鼠标](#)
- [18.3.2 获取鼠标位置](#)
- [18.4 项目：“现在鼠标在哪里？”](#)
- [第1步：导入模块](#)
- [第2步：编写退出代码和无限循环](#)

[第3步：获取并打印鼠标坐标](#)
[18.5 控制鼠标交互](#)
[18.5.1 点击鼠标](#)
[18.5.2 拖动鼠标](#)
[18.5.3 滚动鼠标](#)
[18.6 处理屏幕](#)
[18.6.1 获取屏幕快照](#)
[18.6.2 分析屏幕快照](#)
[18.7 项目：扩展mouseNow程序](#)
[18.8 图像识别](#)
[18.9 控制键盘](#)
[18.9.1 通过键盘发送一个字符串](#)
[18.9.2 键名](#)
[18.9.3 按下和释放键盘](#)
[18.9.4 热键组合](#)
[18.10 复习PyAutoGUI的函数](#)
[18.11 项目：自动填表程序](#)
[第1步：弄清楚步骤](#)
[第2步：建立坐标](#)
[第3步：开始键入数据](#)
[第4步：处理选择列表和单选按钮](#)
[第5步：提交表单并等待](#)
[18.12 小结](#)
[18.13 习题](#)
[18.14 实践项目](#)
[18.14.1 看起来很忙](#)
[18.14.2 即时通信机器人](#)
[18.14.3 玩游戏机器人指南](#)
[附录A 安装第三方模块](#)
[A.1 pip工具](#)
[A.2 安装第三方模块](#)
[附录B 运行程序](#)
[B.1 第一行](#)
[B.2 在Windows上运行Python程序](#)
[B.3 在OS X和Linux上运行Python程序](#)
[B.4 运行Python程序时禁用断言](#)
[附录C 习题答案](#)
[欢迎来到异步社区！](#)
[异步社区的来历](#)
[社区里都有什么？](#)
[购买图书](#)
[下载资源](#)
[与作译者互动](#)
[灵活优惠的购书](#)
[纸电图书组合购买](#)
[社区里还可以做什么？](#)
[提交勘误](#)
[写作](#)
[会议活动早知道](#)
[加入异步](#)