

Python学习笔记

从入门到实战

张学建◎编著



精讲学习视频全程相伴，扫码即看

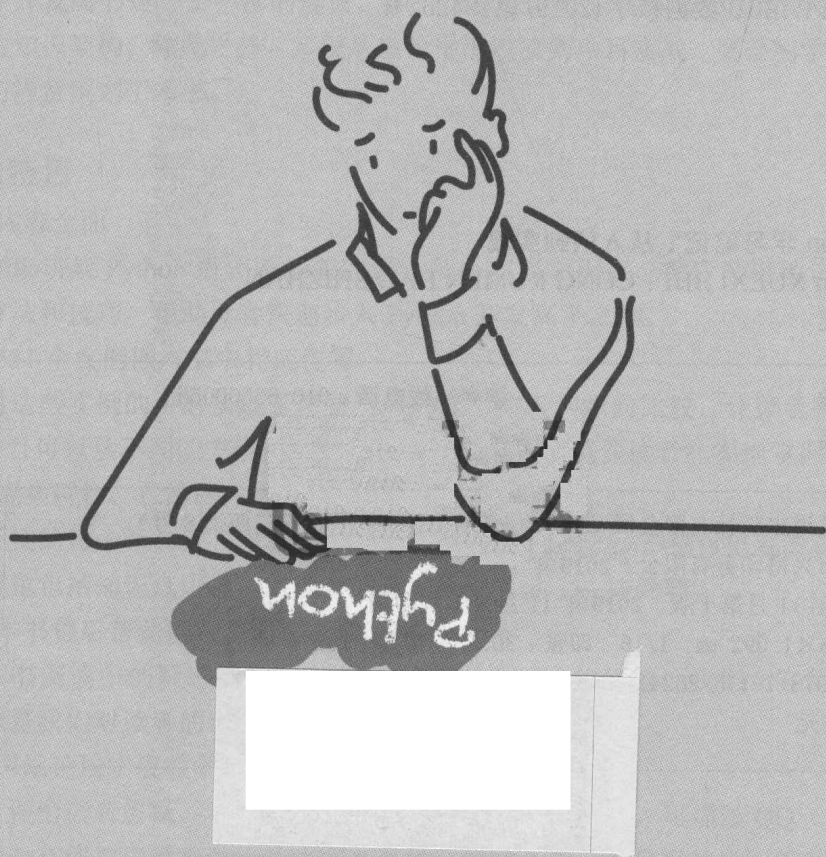
331个有趣实例及精彩分析娓娓道来

中国铁道出版社有限公司
CHINA RAILWAY PUBLISHING HOUSE CO., LTD.

Python学习笔记

从入门到实战

张学建◎编著



中国铁道出版社有限公司
CHINA RAILWAY PUBLISHING HOUSE CO., LTD.

内 容 简 介

本书以学习笔记的形式，循序渐进地讲解了Python语言的核心知识，并通过具体实例的实现过程讲解了各个知识点的使用方法和流程。全书简洁而不失其技术深度，内容丰富全面，更有大量经典案例嵌入书中相应位置。并且本书易于阅读，以极简的文字介绍了复杂的案例，同时涵盖了其他同类图书中很少涉及的历史参考资料，对于初级读者系统学习Python语言大有帮助。

本书旨在帮助有较少编程经验的入门读者系统学习Python语言并通过书中的大量案例学习达到熟练掌握基础开发技能的目的；除此之外，书中还较为完善的纳入了多个综合案例，可帮助进阶类读者透彻理解知识点在具体应用的开发流程，梳理思路，积累开发经验。

图书在版编目（CIP）数据

Python学习笔记：从入门到实战/张学建编著. —北京：中国铁道出版社有限公司，2019.11
ISBN 978-7-113-26241-9

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

中国版本图书馆CIP数据核字(2019)第194262号

书 名：Python 学习 笔记：从入门到实战

Python XUEXI BIJI : CONG RUMEN DAO SHIZHAN

作 者：张学建

责任编辑：荆 波

读者热线电话：010-63560056

责任印制：赵星辰

封面设计：MX DESIGN STUDIO

出版发行：中国铁道出版社有限公司（100054，北京市西城区右安门西街8号）

印 刷：三河市兴博印务有限公司

版 次：2019年11月第1版 2019年11月第1次印刷

开 本：787 mm×1 092 mm 1/16 印张：30.25 字数：736 千

书 号：ISBN 978-7-113-26241-9

定 价：69.80 元

版权所有 侵权必究

凡购买铁道版图书，如有印制质量问题，请与本社读者服务部联系调换。电话：(010) 51873174

打击盗版举报电话：(010) 51873659

从你开始学习编程的那一刻起，就注定了以后所要走的路：从编程学习者开始，依次经历实习生、程序员、软件工程师、架构师、CTO 等职位的磨砺；当你站在职位顶峰的位置蓦然回首，会发现自己的成功并不是偶然，在程序员的成长之路上会有不断修改代码、寻找并解决 Bug、不停测试程序和修改项目的经历；不可否认的是，只要你在自己的开发生涯中稳扎稳打，并且善于总结和学习，最终将会得到可喜的收获。

■ 选择一本合适的书

对于一名程序开发初学者来说，究竟如何学习并提高自己的开发能力呢？选购一本适合自己的程序开发图书是一个不错的建议。如何选择呢，首先这本书要细致到位的帮助自己搭建起基本的知识架构，除此之外，实现从理论平滑过渡到项目实战，要是初学者的迫切需要，为此，我们特意策划了本书。

■ 本书的特色

（1）内容全面

本书详细讲解 Python 语言所涵盖的绝大部分实用知识点，循序渐进地讲解了这些知识点的使用方法和技巧，帮助读者快速步入 Python 开发高手之列。

（2）331 个实例嵌入书中相应位置

通过对这些实例的讲解实现了对知识点的横向切入和纵向比较，让读者有更多的实践演练机会，并且可以从不同的方位展现一个知识点的用法，确保读者扎实地掌握每一个知识点。

（3）视频讲解，二维码布局全书

本书正文的每一个二级目录都有一个二维码，通过二维码扫描可以观看讲解视频，既包括实例讲解也包括知识点讲解，对读者的开发水平实现了拔高处理。

（4）本书售后帮助读者快速解决学习问题

无论本书阅读中的疑惑，还是在 Python 语言学习中的问题，我们会在第一时间为读者解答，这也是我们对读者的承诺。

（5）网站论坛实现教学互动，形成互帮互学的朋友圈

为了方便给读者答疑，我们特为您提供网站论坛技术支持，可通过 QQ（729017304）获得，并且随时在线与读者互动。让大家在互学互帮中形成一个良好的学习编程的氛围。

■ 本书的内容

本书循序渐进的讲解了使用 Python 语言的核心知识，并通过具体实例的实现过程讲解了各个知识点的使用方法和流程。全书共 23 章，分别讲解了 Python 语言基础，Python 基础语法，运算符和表达式，列表、元组和字典，流程控制语句，函数，面向对象编程技术，模块、包和迭代器，生成器、装饰器和闭包，文件操作处理，标准库函数，异常处理，正则表达式，开发网络程序，多线程开发，Tkinter 图形化界面开发，开发数据库程序，使用 Pygame 开发游戏，Python Web 开发，数据可视化，Python 多媒体开发实战，开发网络爬虫，大数据实战：网络爬虫房价数据并数据分析。全书简洁而不失其技术深度，内容丰富全面，历史资料翔实齐全。并且本书易于阅读，以极简的文字介绍了复杂的案例，同时涵盖了其他同类图书中很少涉及的历史参考资料，是学习 Python 数据分析的优选教程。

■ 本书的读者对象

翔实的知识点讲解和精练的案例搭配是本书结构和行文的特点，对于有着基本编程常识的读者可通入该种方式迅速搭建起语法架构和功能轮廓，并通过其中的嵌入案例了解知识点到实践的应用渠道。

对于已经从事并了解 Python 编程工作的读者，可通过书中的综合案例梳理开发思路，积累实践经验。

■ 致谢

本书在编写过程中，得到了中国铁道出版社有限公司编辑的大力支持，正是各位编辑的求实，耐心和效率，才使得本书能够在这么短的时间内出版。另外，也十分感谢我的家人给予的巨大支持。本人水平毕竟有限，书中存在纰漏之处在所难免，诚请读者提出宝贵的意见或建议，以便修订并使之日臻完善。

最后感谢您购买本书，希望本书能成为您编程路上的领航者，祝您阅读快乐！

编者

2019 年 8 月

第 1 章 Python 语言基础

- 1.1 Python 语言横空出世 1
 - 1.1.1 编程世界的“琅琊榜” 1
 - 1.1.2 Python 语言的突出优势 2
 - 1.1.3 Python 语言的特点 2
- 1.2 安装 Python 运行环境 3
 - 1.2.1 选择版本 3
 - 1.2.2 在 Windows 系统中下载并安装 Python 4
 - 1.2.3 在 MacOS 系统中下载并安装 Python 5
 - 1.2.4 在 Linux 系统中下载并安装 Python 6
- 1.3 使用 IDLE 开发 Python 程序 7
 - 1.3.1 IDLE 介绍 7
 - 1.3.2 使用 IDLE 开发第一个 Python 程序 8
 - 实例 1-1: 输出显示一段文本信息 8
- 1.4 使用 PyCharm 开发 Python 程序 10
 - 1.4.1 下载、安装并设置 PyCharm 10
 - 1.4.2 使用 PyCharm 创建并运行一个 Python 程序 13
 - 实例 1-2: 输出显示“Hello 我们是 TFBOYS 组合!” 13
 - 1.4.3 PyCharm 常用功能介绍 16

第 2 章 Python 基础语法

- 2.1 独一无二缩进规则 18
 - 2.1.1 Python 缩进的严格要求 18
 - 2.1.2 一段使用缩进的 Python 程序 19
 - 实例 2-1: 输出显示“Hello, 欢迎来到 Python 世界!” 19
- 2.2 注释是个好帮手 20
 - 实例 2-2: Python 注释的演示使用 21

2.3	编码要用心学好	21
2.3.1	字符编码	21
2.3.2	Unicode 编码和 UTF-8 编码	22
2.3.3	Python 中的编码	23
2.4	标识符和关键字	24
2.5	变量就是一个存储空间	25
	实例 2-3: 打印变量的值	26
	实例 2-4: 同时给多个变量赋值	26
2.6	输入和输出	27
2.6.1	输入信息	27
	实例 2-5: 获取用户输入的内容	27
2.6.2	输出信息	27
	实例 2-6: 使用函数 print() 输出信息	28
2.7	字符串	29
2.7.1	Python 字符串基础	29
2.7.2	获取字符串中的值	30
	实例 2-7: 打印输出字符串中的值	30
2.7.3	修改字符串	30
	实例 2-8: 修改字符串中的某个值	30
2.7.4	使用转义字符	31
	实例 2-9: 打印带有转义字符的内容	31
2.7.5	格式化显示字符串	32
	实例 2-10: 打印不同格式的字符串	32
2.7.6	使用字符串处理函数	32
	实例 2-11: 打印不同格式的 “I love you”	33
2.8	数字类型	34
2.8.1	整数类型: int	34
2.8.2	浮点型	34
2.8.3	布尔型	35
2.8.4	复数型	36
	实例 2-12: 获取并显示各个变量的类型	36
2.9	数字类型转换	36
2.9.1	内置类型转换函数	37
	实例 2-13: 使用函数 int() 实现整型转换	37
2.9.2	类型转换综合演练	37
	实例 2-14: 实现各种常见类型转换操作	38

第 3 章 运算符和表达式

3.1	运算符和表达式介绍	39
3.2	算术运算符和算术表达式	40
	实例 3-1: 使用算术运算符解答简单数学题	40
3.3	比较运算符和比较表达式	41
3.3.1	比较运算符和比较表达式介绍	41
3.3.2	使用比较运算符和比较表达式	41
	实例 3-2: 比较两个整数	41
3.4	赋值运算符和赋值表达式	42
3.4.1	基本赋值运算符和表达式	42
	实例 3-3: 新赛季勇士的场均目标得分和场均三分得分	42
3.4.2	复合赋值运算符和表达式	43
	实例 3-4: 对两个整数进行赋值处理	43
3.5	位运算符和位表达式	44
3.5.1	位运算符和位表达式介绍	44
3.5.2	使用位运算符和位表达式	45
	实例 3-5: 对整数进行位运算操作	45
3.6	逻辑运算符和逻辑表达式	45
3.6.1	逻辑运算符和逻辑表达式介绍	46
3.6.2	使用逻辑运算符和逻辑表达式	46
	实例 3-6: 对两个整数进行逻辑运算操作	46
3.7	成员运算符和成员表达式	47
3.7.1	成员运算符和成员表达式介绍	47
3.7.2	使用成员运算符和成员表达式	47
	实例 3-7: 判断某整数是否属于列表成员	47
3.8	身份运算符和身份表达式	48
3.8.1	身份运算符和身份表达式介绍	48
3.8.2	使用身份运算符和身份表达式	48
	实例 3-8: 判断 a 和 b 是否有相同的标识	48
3.9	运算符的优先级	49
3.9.1	Python 运算符的优先级介绍	49
3.9.2	使用 Python 运算符的优先级	50
	实例 3-9: 组合使用运算符	50

第 4 章 列表、元组和字典

4.1	列表是最基本的数据结构	51
4.1.1	列表的基本用法	51

实例 4-1: 创建一个名为 girl 的简单列表	51
实例 4-2: 创建了一个包含 3 个数字的列表	52
实例 4-3: 访问并显示列表中元素的值	52
实例 4-4: 使用列表中的值创建信息	53
4.1.2 更新 (修改) 列表中的元素	53
实例 4-5: 修改 girl 列表中某个元素的值 (“美丽” 修改为 “妖艳”)	53
4.1.3 插入新的元素	53
实例 4-6: 使用方法 insert() 在 girl 列表中添加一个新元素 (温柔)	54
4.1.4 在列表中删除元素	54
实例 4-7: 使用 del 语句删除 girl 列表中的某个元素 (美丽)	54
实例 4-8: 使用方法 pop() 删除 girl 列表中某个元素 (端庄)	55
4.1.5 列表排列	55
实例 4-9: 让列表 girl 中的元素按字母顺序排	55
实例 4-10: 使用方法 sorted() 排列列表 cars 中的元素	56
4.1.6 列表的高级操作	56
实例 4-11: 创建列表 L 并实现截取与拼接操作	57
实例 4-12: 创建 3 个列表并实现嵌套功能	57
4.2 元组是一种特殊的列表	58
4.2.1 创建并访问元组	58
实例 4-13: 创建两个元组 (tup1, tup2) 并访问显示元组元素	58
4.2.2 修改元组	59
实例 4-14: 连接组合两个元组中的值并输出新元组	59
4.2.3 删除元组	59
实例 4-15: 创新元组 tup 后删除整个元组	59
4.2.4 使用内置方法操作元组	59
实例 4-16: 打印元组中的元素	60
4.2.5 将序列分解为单独的变量	60
实例 4-17: 分解元组中的元素	60
4.3 字典: 成员以 “键: 值” 对形式存在的数据类型	61
4.3.1 创建并访问字典	61
实例 4-18: 创建字典 dict 并获取字典中 3 个键的值	62
4.3.2 向字典中添加数据	62
实例 4-19: 创建字典 dict 并向其中添加两个数据	62
4.3.3 修改字典	63
实例 4-20: 修改字典 dict 中 Age 元素的值	63
4.3.4 删除字典中的元素	63
实例 4-21: 删除字典 dict 中的 Name 元素	63
4.3.5 和字典有关的内置函数	63
实例 4-22: 输出字典 dict 中校花的基本资料	64
4.3.6 遍历字典	64
实例 4-23: 返回一个字典中的所有键	64
实例 4-24: 使用方法 values() 遍历字典中的所有值	65

4.4	集合是一个无序不重复元素的序列	65
	实例 4-25: 创建集合 student 并实现各种操作	65
第 5 章 流程控制语句		
5.1	条件语句: 是与不是	67
5.1.1	条件语句介绍	67
5.1.2	最简单的 if 语句	68
	实例 5-1: 使用 if 语句判断所能承受的车票价格	68
5.1.3	使用 if...else 语句	68
	实例 5-2: 使用 if...else 语句判断所能承受的酒店价格	69
5.1.4	使用 if...elif...else 语句	69
	实例 5-3: 酒店入住体验打分系统	70
5.1.5	if 语句的嵌套	70
	实例 5-4: 判断一个数字的大小是否合适	70
5.1.6	实现 switch 语句的功能	71
	实例 5-5: 使用 elif 实现其他语言中的 switch 功能	72
	实例 5-6: 用字典实现其他语言中的 switch 功能	73
5.2	for 循环语句: 全部走一遍	73
5.2.1	基本的 for 循环语句	73
	实例 5-7: 使用 for 循环语句输出英文单词的字母	74
5.2.2	通过序列索引迭代	75
	实例 5-8: 循环输出了列表 fruits 中的元素	75
5.2.3	使用 for... else 循环语句	75
	实例 5-9: 判断是否是质数 (10 ~ 20 之间的数字)	76
5.2.4	嵌套 for 循环语句	76
	实例 5-10: 获取两个整数之间的所有素数	76
5.3	while 循环语句: 不知道重复多少次	77
5.3.1	基本的 while 循环语句	77
	实例 5-11: 使用 while 循环语句显示小鸟的假期	78
5.3.2	使用 while...else 循环语句	78
	实例 5-12: 设置到 5 停止循环	79
5.3.3	死循环问题	79
	实例 5-13: 使用 while 循环时的死循环问题	79
5.3.4	使用 while 循环嵌套语句	80
	实例 5-14: 输出 0 到 100 之内的素数	80
5.4	循环控制语句: 处理突发事件	81
5.4.1	使用 break 语句	81
	实例 5-15: 使用 break 停止循环	82
5.4.2	使用 continue 语句	82
	实例 5-16: 循环输出字母和数字	82

5.4.3 使用 pass 语句	83
实例 5-17: 输出指定字符串 python 中的每个英文字母	83

第 6 章 函数

6.1 函数就是某个指定功能的语句	85
6.1.1 定义函数	85
实例 6-1: 定义了一个基本的输出信息函数	86
6.1.2 调用函数	86
实例 6-2: 定义表调用函数, 计算元组内元素的和	86
6.2 函数的参数	87
6.2.1 形参和实参	87
6.2.2 必需参数	87
实例 6-3: 一个错误用法	88
6.2.3 关键字参数	88
实例 6-4: 打印网友的信息	88
实例 6-5: 在使用函数参数时不需要指定顺序	88
6.2.4 默认参数	89
实例 6-6: 打印两个好友的信息	89
6.2.5 不定长参数	89
实例 6-7: 在同一个函数中分别使用一个参数和三个参数	90
6.2.6 按值传递参数和按引用传递参数	90
实例 6-8: 函数的参数是一个列表	90
6.3 函数的返回值	91
6.3.1 返回一个简单值	91
实例 6-9: 定义函数 get_name() 并通过形参返回一个简单的值	91
6.3.2 可选实参	91
实例 6-10: 让实参变成一个可选参数	91
6.3.3 返回一个字典	92
实例 6-11: 定义函数 person() 并让其返回值是一个字典	92
6.4 变量的作用域	93
6.4.1 三种变量作用域	93
6.4.2 使用变量作用域	93
实例 6-12: 使用相互不影响的同名变量	93
实例 6-13: 使用关键字 global 在函数内部使用全局变量	94
6.5 使用函数传递列表	94
6.5.1 访问列表中的元素	94
实例 6-14: 定义函数 users() 并问候列表中的每个好友	94
6.5.2 在函数中修改列表	95
实例 6-15: 定义函数 copy() 并复制好友到“亲人”分组	95

6.6	lambda 来创建和使用匿名函数	96
6.6.1	匿名函数介绍	96
6.6.2	使用匿名函数	96
	实例 6-16: 使用 lambda 创建匿名函数 sum() 并输出恋爱前后的幸福 指数值	96
6.7	函数和模块开发	97
6.7.1	导入整个模块文件	97
	实例 6-17: 导入模块文件的全部内容	97
6.7.2	只导入指定的函数	98
	实例 6-18: 导入模块文件中的某个函数	98
6.7.3	使用 as 设置函数的别名	98
	实例 6-19: 给函数设置一个别名	99
6.7.4	使用 as 设置模块别名	99
	实例 6-20: 给模块设置一个别名	99
6.7.5	导入所有函数	99
	实例 6-21: 使用 “*” 运算符导入外部模块文件中所有函数	100
第 7 章	面向对象编程技术	
7.1	面向对象基础	101
7.1.1	Python 的面向对象编程	101
7.1.2	Python 面向对象的几个核心概念	102
7.2	定义并使用类	103
7.2.1	定义一个类	103
7.2.2	类的基本用法	104
	实例 7-1: 财产分配说明书	104
7.3	对象	104
	实例 7-2: 使用类的对象	105
7.4	方法	105
7.4.1	定义并使用类方法	105
	实例 7-3: 调用类方法打印信息	105
7.4.2	构造方法	106
	实例 7-4: 使用构造方法打印信息	106
	实例 7-5: 霍老二的宠物狗	107
7.4.3	方法调用	108
	实例 7-6: 在类中调用类的自身方法和全局函数	108
7.4.4	创建多个实例	108
	实例 7-7: 创建多个不同的对象实例	109
7.4.5	使用私有方法	109
	实例 7-8: 打印网店的地址	110
7.4.6	析构方法	110

实例 7-9: 打印三种客户类型信息	110
7.4.7 静态方法和类方法	111
实例 7-10: 同时使用类方法和静态方法	111
7.4.8 类的专有方法	112
7.5 属性	112
7.5.1 认识属性	113
7.5.2 类属性和实例属性	113
实例 7-11: 改变变量的值	114
7.5.3 设置属性的默认值	115
实例 7-12: 设置汽车的属性信息	115
7.5.4 修改属性的值	116
实例 7-13: 通过实例修改霍老板新车的行驶里程	116
实例 7-14: 通过自定义方法修改霍老板新车行驶里程的方法	117
实例 7-15: 通过递增值修改霍老板新车的行驶里程	117
7.5.5 使用私有属性	118
实例 7-16: 打印私有属性的值	118
7.6 继承	118
7.6.1 定义子类	118
实例 7-17: 霍老二的一辆豆腐车	119
7.6.2 在子类中定义方法和属性	120
实例 7-18: 豆腐车的发动机参数	120
7.6.3 子类可以继续派生新类	121
实例 7-19: 豆腐车的升级版	121
7.6.4 私有属性和私有方法	122
实例 7-20: 不能相互访问私有变量	122
7.6.5 多重继承	122
实例 7-21: 3 个子类的多重继承	123
7.7 方法重写	124
实例 7-22: 通过重写修改方法的功能	124

第 8 章 模块、包和迭代器

8.1 模块架构	126
8.1.1 最基本的模块调用	126
实例 8-1: 在程序中调用外部模块文件	126
实例 8-2: 外部模块文件和测试文件不在同一个目录	127
8.1.2 目录 “ <code>__pycache__</code> ”	128
实例 8-3: 编译指定的文件	128
8.1.3 使用 “ <code>__name__</code> ” 属性	129
实例 8-4: 测试模块是否能正常运行	129
8.2 包是管理程序模块的形式	130

8.2.1	表示包	130
8.2.2	创建并使用包	131
8.2.3	创建并使用包	132
	实例 8-5: 创建并使用包输出指定的内容	132
8.3	迭代器: 简化代码并节约内存	133
8.3.1	什么是迭代器	133
	实例 8-6: 使用 for 循环语句遍历迭代器	133
8.3.2	创建并使用迭代器	134
	实例 8-7: 打印迭代的元素	134
8.3.3	使用内置迭代器协议方法 iter()	134
	实例 8-8: 显示迭代器中的数据元素	135
第 9 章 生成器、装饰器和闭包		
9.1	生成器: 边循环边计算	136
9.1.1	生成器的运行机制	136
	实例 9-1: 使用 yield 生成器显示奥运会金牌榜的变化	137
9.1.2	创建生成器	137
	实例 9-2: 创建一个递减序列生成器	137
9.1.3	生成器的第一次调用	139
	实例 9-3: 重新初始化生成器生成初始值	139
9.1.4	使用协程重置生成器序列	139
	实例 9-4: 使用方法 send() 重置生成器的序列	139
9.2	装饰器: 拓展函数功能	140
9.2.1	创建装饰器	140
9.2.2	使用装饰器修饰函数	141
	实例 9-5: 比较装饰器函数和非装饰器函数	141
	实例 9-6: 比较调用前后的执行结果	142
9.2.3	使用装饰器修饰类	142
	实例 9-7: 打印 x、y、z 的坐标	142
9.3	命名空间: 存储变量与值对应关系的字典	143
9.3.1	命名空间的本质	143
	实例 9-8: 函数与其全局命名空间	144
9.3.2	查找命名空间	145
	实例 9-9: 查找嵌套函数命名空间	145
9.3.3	命名空间的生命周期	146
9.3.4	命名空间访问函数 locals() 与 globals()	146
	实例 9-10: 使用内置函数 locals() 访问局部命名空间	146
	实例 9-11: 使用函数 globals() 访问全局命名空间	147
	实例 9-12: 同时使用内置函数 locals 与 globals, 并分析三者区别	147
9.4	闭包: 函数和引用环境组合而成的实体	148

9.4.1	什么是闭包	148
实例 9-13	打印教练的信息	148
9.4.2	闭包和嵌套函数	149
实例 9-14	使用被嵌套函数中的变量	149
9.4.3	使用闭包记录函数被调用的次数	150
实例 9-15	统计调用函数的次数	150

第 10 章 文件操作处理

10.1	文件操作基础	151
10.2	使用 File 操作文件	152
10.2.1	File 对象介绍	152
实例 10-1	打开一个文件并查看其属性	153
10.2.2	使用 close() 方法关闭操作	153
实例 10-2	打开和关闭一个记事本文件	154
10.2.3	使用方法 flush()	154
实例 10-3	使用 flush() 方法刷新缓冲区	154
10.2.4	使用方法 fileno()	154
实例 10-4	使用方法 fileno() 返回文件描述符	154
10.2.5	使用方法 isatty()	155
实例 10-5	检测某文件是否连接到一个终端设备	155
10.2.6	使用方法 next()	155
实例 10-6	返回文件各行内容	156
10.2.7	使用方法 read()	156
实例 10-7	读取文件中 3 个字节的内容	156
10.3	使用 OS 对象	157
10.3.1	OS 对象介绍	157
10.3.2	使用方法 access()	159
实例 10-8	获取指定文件的操作权限	159
10.3.3	使用方法 chdir()	160
实例 10-9	修改当前工作目录到指定路径	160
10.3.4	使用方法 chmod()	160
实例 10-10	修改指定文件或目录权限	161
10.3.5	打开、写入和关闭	161
实例 10-11	创建并打开文件“456.txt”，然后实现文件的打开、写入和关闭操作	162
10.3.6	打开、读取和关闭	163
实例 10-12	读取文件中的指定字符	163
10.3.7	创建目录	163
实例 10-13	使用方法 mkdir() 创建一个目录“top”	164
实例 10-14	使用方法 makedirs() 创建一个目录“tmp/home/123”	164

第 11 章 标准库函数

11.1	字符串处理函数	166
11.1.1	分割字符串	166
	实例 11-1: 使用函数 <code>split()</code> 分割指定的字符串	166
	实例 11-2: 使用函数 <code>re.split()</code> 分割指定字符串	167
11.1.2	字符串开头和结尾处理	167
	实例 11-3: 使用函数 <code>startswith()</code> 和 <code>endswith()</code> 处理指定字符串	168
11.1.3	实现字符串匹配处理	168
	实例 11-4: 匹配处理各种类型的字符	169
11.1.4	文本模式匹配和查找	170
	实例 11-5: 查找字符串中字符	170
11.1.5	文本查找和替换	170
	实例 11-6: 使用函数 <code>replace()</code> 替换文本	170
11.1.6	实现最短文本匹配	171
	实例 11-7: 使用正则表达式实现最短文本匹配	171
11.1.7	处理 Unicode 文本	171
	实例 11-8: Unicode 字符串的代码点序列表示方法	171
	实例 11-9: 使用函数 <code>normalize()</code> 归一化 Unicode 字符串	172
11.1.8	删除字符串中的字符	172
	实例 11-10: 删除字符串中的指定的内容	173
11.1.9	字符过滤和清理	174
	实例 11-11: 删除空格和 Unicode 组合字符	174
11.1.10	字符串对齐处理	175
	实例 11-12: 按照指定格式对齐字符串	176
11.1.11	字符串连接和合并	176
	实例 11-13: 合并给出的多个字符串	176
11.2	数字处理函数	177
11.2.1	使用 <code>math</code> 模块实现数学运算	178
	实例 11-14: 使用函数 <code>abs()</code> 返回数字绝对值	178
	实例 11-15: 使用函数 <code>ceil()</code> 返回最小整数	178
	实例 11-16: 使用两种格式的 <code>pow()</code> 函数	180
	实例 11-17: 计算指定数字四舍五入值	180
11.2.2	使用 <code>decimal</code> 模块实现精确运算	181
	实例 11-18: 分别实现误差运算和精确运算	181
11.2.3	处理二进制、八进制和十六进制数据	182
	实例 11-19: 将一个整数转换为二进制、八进制或十六进制	182
11.2.4	实现复数运算	183
	实例 11-20: 操作处理复数数据	183
11.2.5	使用 <code>fractions</code> 模块处理分数	184
	实例 11-21: 对分数进行各种运算处理	185
11.2.6	使用 <code>NumPy</code> 模块	185
	实例 11-22: 创建一维数组和二维数组	185

11.3	日期和时间函数	186
11.3.1	使用时间模块	186
	实例 11-23: 使用函数 <code>time.clock()</code> 实现时间处理	186
	实例 11-24: 使用函数 <code>mktime(tupletime)</code> 实现时间操作	187
	实例 11-25: 使用函数 <code>time.tzset()</code> 格式化显示时间	188
11.3.2	使用 <code>Calendar</code> 日历模块	189
	实例 11-26: 使用 <code>calendar</code> 模块函数显示日历	190
11.3.3	使用 <code>datetime</code> 模块	191
	实例 11-27: 使用类 <code>date</code> 打印系统日期信息	192
	实例 11-28: 使用类 <code>date</code> 打印不同格式的日期	193
	实例 11-29: 使用类 <code>time</code> 打印时间信息	194
	实例 11-30: 使用类 <code>datetime</code> 打印系统时间信息	195
第 12 章	异常处理	
12.1	语法错误	196
	实例 12-1: 代码拼写错误的执行输出	196
12.2	常见异常处理方式	197
12.2.1	异常的特殊之处	197
12.2.2	使用 “ <code>try...except</code> ” 处理异常	198
	实例 12-2: 处理变量错误	199
	实例 12-3: 一个 <code>try</code> 语句包含多个 <code>except</code> 子句的异常处理	199
12.2.3	使用 “ <code>try...except...else</code> ” 处理异常	199
	实例 12-4: 处理索引异常	200
12.2.4	使用 “ <code>try...except...finally</code> ” 语句	201
	实例 12-5: 使用 <code>finally</code> 确保使用文件后能关闭这个文件	201
12.3	抛出异常	202
12.3.1	使用 <code>raise</code> 抛出异常	202
	实例 12-6: 抛出错误异常	202
12.3.2	使用 <code>assert</code> 语句抛出异常	203
	实例 12-7: 抛出循环错误异常	203
12.3.3	自定义异常	203
	实例 12-8: 自己编写一个异常类	204
12.4	内置异常类	204
12.4.1	处理 <code>ZeroDivisionError</code> 异常	205
	实例 12-9: 解决 <code>ZeroDivisionError</code> 类型的异常	205
12.4.2	<code>FileNotFoundError</code> 异常	206
	实例 12-10: 解决 <code>FileNotFoundError</code> 类型的异常	206
12.4.3	使用 <code>except</code> 捕获异常	207
	实例 12-11: 捕获程序中的所有异常	207
12.4.4	逻辑错误的程序测试	207
	实例 12-12: 使用函数 <code>testmod()</code> 实现单元测试	208