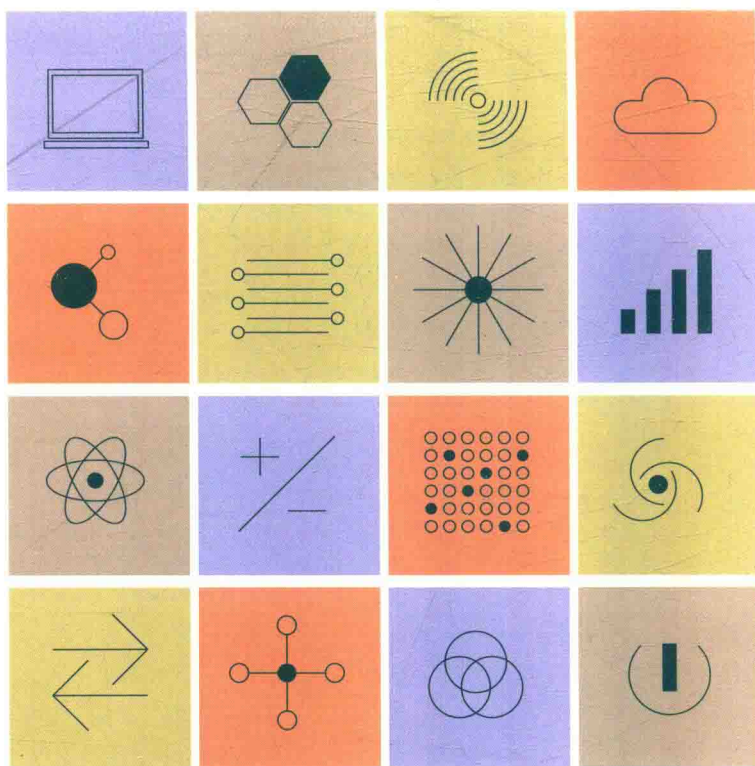


- Python程序设计从小白到大牛
- 理论知识与编程任务双主线学习
- 102个实例+79个任务+多个二维码拓展学习资料

# Python 郑凯梅 编著

# 程序设计

## 任务驱动式教程



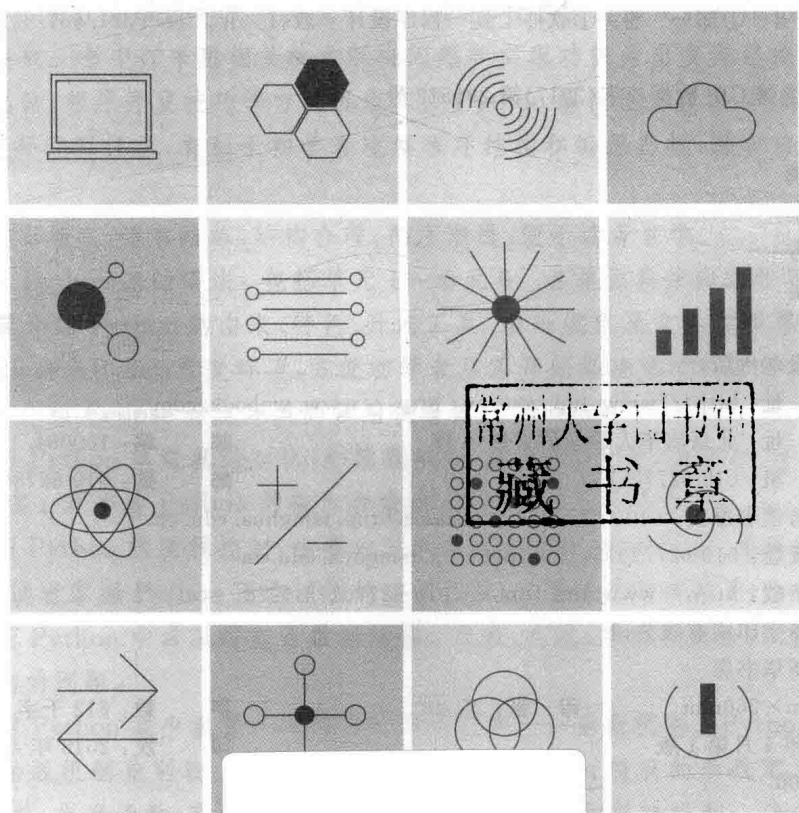
清华大学出版社

# Python

郑凯梅◎编著

# 程序设计

# 任务驱动式教程



清华大学出版社  
北京

## 内 容 简 介

本书对 Python 程序设计的教学内容进行了系统化设计,形成了具有 3 个学习阶段(Python 基础语法、Python 面向对象程序设计和 Python 高级应用)、14 个单元和 2 条主线(理论知识主线和编程任务主线)的体系结构。

本书内容翔实,结构合理,语言精练,表述清晰,实用性强,易于自学,主要内容包括搭建环境和运行 Python 应用程序、Python 基本语法、Python 流程控制、列表与元组、函数与模块、文件、面向对象编程、异常处理、GUI 编程、进程和线程、Python 与数据库、网络编程、Web 编程、Python 工程应用等。

本书适合作为高等学校计算机及其他理工科类专业的教材,也可以作为相关培训机构的培训教材,还可以供对 Python 程序设计感兴趣的广大读者自学时选用。

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 13701121933

### 图书在版编目(CIP)数据

Python 程序设计任务驱动式教程/郑凯梅编著. —北京:清华大学出版社,2018  
ISBN 978-7-302-49046-3

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

中国版本图书馆 CIP 数据核字(2017)第 295512 号

责任编辑:刘翰鹏

封面设计:傅瑞学

责任校对:袁 芳

责任印制:丛怀宇

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质量反馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

课件下载: <http://www.tup.com.cn>, 010-62770175-4278

印 装 者:北京泽宇印刷有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:25.25

字 数:612 千字

版 次:2018 年 4 月第 1 版

印 次:2018 年 4 月第 1 次印刷

印 数:1~2000

定 价:59.00 元

产品编号:074912-01



Python 是一种跨平台的面向对象的程序设计语言,具有简单性、易学性、开源性、可移植性、可扩展性和丰富类库支持的特点,是目前非常流行的程序设计语言之一,广泛应用于窗口界面程序开发、网络程序开发、数据库程序开发、嵌入式程序开发和机器学习开发等。

本书特色如下所述。

(1) 本书对 Python 程序设计的教学内容进行了系统化设计,形成了 Python 基础语法、Python 面向对象程序设计和 Python 高级应用 3 个学习阶段、14 个单元的体系结构。

(2) 每个教学单元由理论知识、案例和任务组成。其中,理论知识和案例相融合,便于读者掌握基本编程思想和语法;任务部分将相关知识点综合应用,通过这一环节的训练,提高读者分析问题和解决问题的能力,达到学以致用目标。

(3) 任务环节由任务描述和任务实现组成。其中,任务实现由设计思路、源代码清单和程序运行结果组成。书中以采用相关技术解决问题和实现功能为出发点组织任务环节,让读者通过编程思路、程序开发技巧等方面逐步掌握 Python 编程相关知识,提高编程能力。

(4) 程序代码注释详尽,有利于初学者理解程序结构和编程思想,既有启发性,又降低了学习难度。

(5) 本书内容翔实,语言精练,结构合理,循序渐进,便于读者自学。

初级篇——Python 基础语法:包括单元 1~单元 6。各单元具体内容如下所述。

单元 1 简要介绍 Python 的由来、特色、开发工具、编码规范及文件类型等方面,详细介绍如何搭建 Eclipse+Pydev 开发环境,方便初学者从零开始搭建环境。最后讲解如何开发 Python 程序,并介绍 Eclipse 开发环境的常用快捷键。

单元 2 介绍 Python 编程基础知识,如数据类型、标识符、变量、运算符、字符串、正则表达式、数学运算等;还讲述 Python 的输入和输出,为开发程序做好准备。

单元 3 介绍 Python 的流程控制,主要内容包括顺序结构、选择结构、循环结构及循环结构的退出,帮助读者掌握 Python 面向过程的编程技术,并能设计简单的 Python 程序。

单元 4 介绍 Python 中常用的内置数据结构:列表、元组、字典和集合,以便读者解决一些复杂存储结构的问题。

单元 5 介绍 Python 减少重复代码编写的解决机制——函数机制。Python 的函数机制与其他语言的函数机制差别较大,本单元详细阐述了 Python 特有的参数定义、参数传递、返回值、匿名函数、嵌套函数、高级函数、递归函数等,以及包和模块机制。读者可以根据实际情况灵活地选用适当的函数或模块机制来解决问题。

单元 6 介绍 Python 文件、目录和 CSV 文件的操作,以便读者对文本文件、二进制文件及其他类型的文件,如电子表格文件等进行输入和输出操作。

中级篇——Python 面向对象程序设计:包括单元 7~单元 10。各单元具体内容如下

所述。

单元7介绍Python实现面向对象编程设计中的类、继承、多态、抽象类等的技术,以便读者使用面向对象的技术来解决问题。

单元8介绍Python的异常处理机制和断言机制,包括异常处理、捕获异常、抛出异常等,以便读者在高级程序设计中正确处理Python程序中出现的异常和错误。

单元9介绍Python图形界面开发库Tkinter模块和核心功能,包括界面布局、常用控件、对话框等,以便读者利用Tkinter模块提供的控件开发完整的、功能完备的GUI应用程序。

单元10介绍Python的多线程和多进程机制,包括多线程、多进程、线程之间的同步等技术,以便读者编程解决并发类的问题。

高级篇——Python高级应用:包括单元11~单元14。各单元具体内容如下所述。

单元11介绍Python的数据库编程接口,主要讲述SQLite和MySQL数据库的操作方法,以便读者完成嵌入式数据库应用或信息管理类应用程序的开发。

单元12介绍Python网络编程,包括Socket客户端和服务端编程、SocketServer编程、多连接应用、FTP、电子邮件的接收和发送等,以便读者轻松开发通信类程序。

单元13介绍Python开发Web应用程序,包括普通Web表单程序设计、Tornado服务器和SQLAlchemy模块,以便读者开发MVC模式的B/S结构的应用程序。

单元14介绍Python工程应用,包括NumPy、SciPy、Matplotlib库的应用,以便读者解决一些工程应用问题或绘制各类图表。

为满足教学和读者的需要,本书配有电子课件以及书中示例源码。需要者,请到清华大学出版社网站<http://www.tup.com.cn/>下载。

由于编者水平有限,书中难免存在不足之处,恳请读者批评、指正。

编者

2017年10月

# 目 录



## 初级篇——Python 基础语法

单元 1 搭建环境和运行 Python 应用程序 .....	003
1.1 认识 Python .....	003
1.1.1 Python 的由来 .....	003
1.1.2 Python 的特色 .....	004
1.1.3 Python 的开发工具 .....	004
1.1.4 Python 文件类型 .....	005
1.1.5 Python 编码规范 .....	006
1.2 Python 程序开发环境的搭建与配置 .....	007
任务 1-1 Python 程序开发环境的搭建与配置 .....	007
1.3 Eclipse 集成开发环境使用 .....	007
任务 1-2 编写第一个程序 Hello World .....	007
1.4 习题 .....	010
单元 2 Python 基础语法 .....	011
2.1 基本数据类型 .....	011
2.1.1 数值 .....	011
2.1.2 字符串 .....	012
2.1.3 变量 .....	016
任务 2-1 信息查找 .....	016
2.1.4 正则表达式 .....	017
任务 2-2 电子邮箱格式检测 .....	019
任务 2-3 电话号码检测 .....	020
2.2 运算符与表达式 .....	021
2.2.1 算术运算符与算术表达式 .....	021
任务 2-4 计算圆锥体的体积和表面积 .....	022
2.2.2 关系运算符和逻辑运算符 .....	023
任务 2-5 闰年判断 .....	024
2.2.3 赋值运算符 .....	025
2.2.4 位运算符 .....	025
2.2.5 成员运算符 .....	026



2.3	Python 输入 .....	026
2.4	Python 输出 .....	027
	任务 2-6 位运算实例 .....	028
2.5	Python 数学运算 .....	029
2.5.1	分数 .....	029
	任务 2-7 分数运算 .....	030
2.5.2	复数 .....	030
	任务 2-8 复数运算 .....	031
2.5.3	math 模块 .....	031
	任务 2-9 计算汽车贷款 .....	031
2.6	习题 .....	032
<b>单元 3</b>	<b>Python 流程控制 .....</b>	<b>034</b>
3.1	顺序结构 .....	034
	任务 3-1 计算椭球的表面积和体积 .....	034
3.2	选择结构 .....	035
3.2.1	if-else 条件语句 .....	035
	任务 3-2 输出最大的数 .....	036
	任务 3-3 计算一元二次方程的根 .....	037
3.2.2	if-elif-else 判断语句 .....	038
	任务 3-4 成绩分等 .....	038
3.2.3	if 语句的嵌套 .....	039
	任务 3-5 判断三角形的类型 .....	040
3.2.4	switch 语句的替代方案 .....	041
	任务 3-6 简单的计算器 .....	041
3.3	循环结构 .....	042
3.3.1	while 循环 .....	042
	任务 3-7 自然数求和 .....	043
	任务 3-8 计算圆周率 .....	044
	任务 3-9 系列数据的统计 .....	045
3.3.2	for 循环语句 .....	046
	任务 3-10 计算 $n$ 的阶层 .....	047
	任务 3-11 计算分数之和 .....	048
3.3.3	break 和 continue 语句 .....	049
	任务 3-12 素数判断 .....	049
	任务 3-13 用户登录模拟 .....	050
	任务 3-14 数值计算 .....	051
3.3.4	循环中的 else 语句 .....	052
	任务 3-15 输出素数 .....	053
3.3.5	嵌套循环 .....	054



任务 3-16 输出九九乘法表 .....	054
任务 3-17 输出水仙花数 .....	055
3.3.6 字符串的遍历循环 .....	055
任务 3-18 创建扑克牌 .....	056
任务 3-19 字符串逆序输出 .....	057
3.3.7 pass 语句 .....	057
3.4 习题 .....	058
<b>单元 4 列表与元组 .....</b>	<b>059</b>
4.1 列表 .....	059
4.1.1 列表的创建和使用 .....	059
任务 4-1 学生成绩统计 .....	064
任务 4-2 学生信息管理 .....	066
4.1.2 列表解析 .....	070
任务 4-3 输出乘法表 .....	072
4.1.3 列表实现堆栈 .....	073
任务 4-4 表达式括号匹配 .....	073
4.1.4 列表实现队列 .....	074
任务 4-5 约瑟夫环问题 .....	075
4.2 元组 .....	076
4.2.1 元组的创建和使用 .....	076
4.2.2 不可变和可变对象 .....	077
任务 4-6 扑克游戏发牌模拟 .....	077
4.3 字典 .....	079
4.3.1 创建和使用字典 .....	079
任务 4-7 字符个数统计 .....	082
4.3.2 管理字典 .....	083
任务 4-8 用户注册与登录模拟 .....	083
4.4 集合 .....	084
4.4.1 集合的创建和使用 .....	085
任务 4-9 集合运算小测验 .....	086
4.4.2 集合运算 .....	087
任务 4-10 简单的购物分析 .....	088
4.5 习题 .....	090
<b>单元 5 函数与模块 .....</b>	<b>091</b>
5.1 函数 .....	091
5.1.1 函数的定义与使用 .....	091
任务 5-1 爱心输出 .....	092
5.1.2 函数的参数 .....	093
任务 5-2 输出指定范围内的素数 .....	097





5.1.3	函数的返回值	098
任务 5-3	关键字检索	099
5.1.4	变量作用域	101
5.1.5	匿名函数	103
任务 5-4	两个整数的位运算	104
5.1.6	高阶函数	105
5.1.7	函数的嵌套	107
任务 5-5	矩阵相乘	108
5.1.8	递归函数	110
任务 5-6	二分查找算法的递归实现	110
5.2	模块	112
5.2.1	模块的创建	112
5.2.2	导入模块	114
5.2.3	包	118
5.2.4	常用的内置模块	118
5.2.5	第三方模块	118
5.3	习题	119
<b>单元 6</b>	<b>文件</b>	<b>120</b>
6.1	文件的操作	120
6.1.1	文件的打开和关闭	120
6.1.2	读文件	122
任务 6-1	文件比较	124
6.1.3	写文件	125
任务 6-2	文件分割与合并	126
6.1.4	文件的其他操作	130
6.1.5	pickle 模块	130
任务 6-3	四则运算练习系统	131
6.2	目录的操作	134
6.2.1	目录与文件操作函数	134
任务 6-4	图片文件批量重命名	135
6.2.2	目录的遍历	137
任务 6-5	批量修改所有文件名为小写	139
6.3	CSV 文件	140
6.3.1	CSV 文件简介	140
6.3.2	CSV 文件访问	141
任务 6-6	读取 CSV 文件中指定行或列的数据	142
6.3.3	Excel 文件与 CSV 文件	144
任务 6-7	Excel 文件与 CSV 文件的相互转换	144
6.4	习题	146



## 中级篇——Python 面向对象程序设计

单元 7 面向对象编程 .....	151
7.1 面向对象概述 .....	151
7.2 类和对象 .....	152
7.2.1 类的定义和对象的创建 .....	152
7.2.2 实例变量及封装 .....	153
7.2.3 方法 .....	155
7.2.4 属性方法 .....	156
7.2.5 类中的其他内置方法和属性 .....	158
任务 7-1 简单的购物车管理 .....	159
7.3 继承 .....	165
7.3.1 使用继承 .....	165
任务 7-2 单继承与多继承实例 .....	167
7.3.2 抽象基类 .....	171
任务 7-3 抽象类应用 .....	174
7.3.3 多态性 .....	176
7.4 运算符的重载 .....	177
任务 7-4 复数运算 .....	177
7.5 习题 .....	180
单元 8 异常处理 .....	181
8.1 Python 中的异常 .....	181
8.2 常用异常处理 .....	181
8.3 自定义异常 .....	184
任务 8-1 银行转账处理模拟 .....	185
8.4 习题 .....	188
单元 9 GUI 编程 .....	189
9.1 GUI 程序开发简介 .....	189
9.2 Tkinter 包 .....	189
9.2.1 Tkinter 包简介 .....	189
9.2.2 创建 GUI 应用程序 .....	191
9.2.3 Tkinter 布局管理 .....	195
任务 9-1 猜数字游戏 .....	199
9.3 Tkinter 控件 .....	202
9.3.1 Widget 控件 .....	202
9.3.2 Label 控件 .....	204
9.3.3 Entry 控件 .....	205
9.3.4 Button 控件 .....	206
9.3.5 Frame 控件 .....	206



任务 9-2 登录界面模拟 .....	207
9.3.6 Radiobutton 控件 .....	209
9.3.7 Checkbutton 控件 .....	212
任务 9-3 简单的测试系统 .....	214
9.3.8 Listbox 控件 .....	218
任务 9-4 信息填写与反馈 .....	220
9.3.9 菜单控件 .....	223
任务 9-5 记事本 .....	226
9.4 对话框 .....	234
9.4.1 标准对话框 .....	235
9.4.2 自定义对话框 .....	238
任务 9-6 选择和替换对话框 .....	238
9.5 习题 .....	242
<b>单元 10 进程和线程 .....</b>	<b>244</b>
10.1 Python 下的进程编程 .....	244
10.1.1 创建进程 .....	244
10.1.2 进程池 .....	246
10.1.3 多进程间通信 .....	249
任务 10-1 多进程实现大文件分割 .....	253
10.2 多线程编程 .....	256
任务 10-2 多线程下载网络文件 .....	260
10.3 线程之间的同步 .....	263
10.3.1 锁机制 .....	263
10.3.2 条件变量机制 .....	266
10.3.3 队列机制 .....	268
10.3.4 事件机制 .....	270
任务 10-3 URL 请求 .....	270
10.4 习题 .....	273

## 高级篇——Python 高级应用

<b>单元 11 Python 与数据库 .....</b>	<b>277</b>
11.1 Python 数据库编程接口 .....	277
11.1.1 全局变量 .....	277
11.1.2 异常处理 .....	278
11.1.3 数据库连接与游标 .....	278
11.1.4 数据类型 .....	279
11.1.5 Python 数据库操作步骤 .....	279
11.2 SQLite 数据库操作 .....	280
11.2.1 SQLite 数据库连接 .....	280

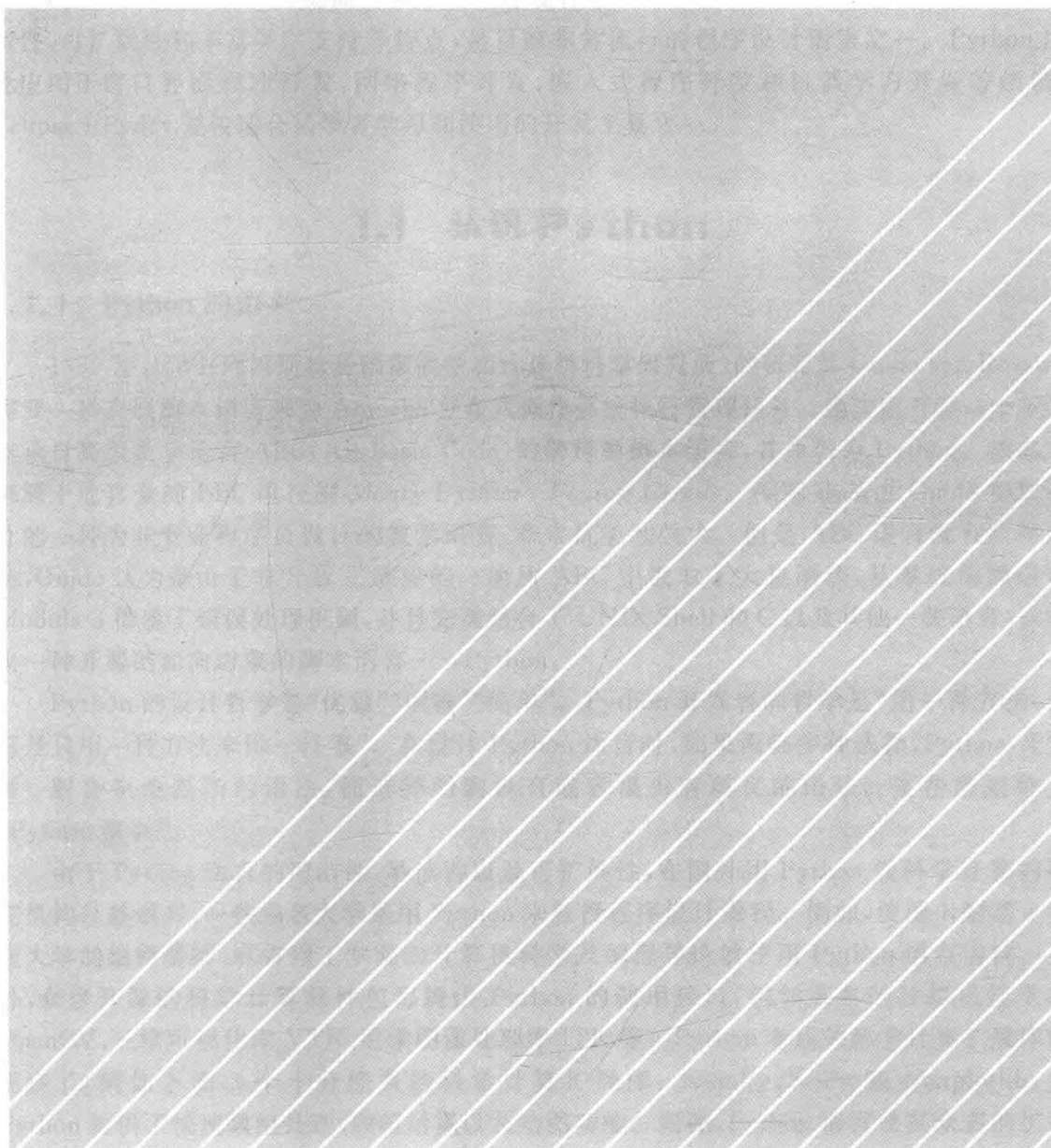
11.2.2	SQLite 数据库操作步骤 .....	281
任务 11-1	通讯录管理系统 .....	281
11.3	MySQL 数据库操作 .....	286
11.3.1	MySQL 数据库连接 .....	286
11.3.2	MySQL 数据库操作步骤 .....	287
任务 11-2	ATM 电子银行模拟 .....	287
11.4	习题 .....	306
<b>单元 12</b>	<b>网络编程 .....</b>	<b>307</b>
12.1	socket 模块 .....	307
12.1.1	socket 模块介绍 .....	307
12.1.2	网络客户端 .....	309
12.1.3	网络服务器 .....	311
任务 12-1	局域网文件传输 .....	313
12.2	SocketServer 模块 .....	317
任务 12-2	基于 SocketServer 的文件上传 .....	318
12.3	多连接应用 .....	322
12.3.1	使用 SocketServer 进行多连接处理 .....	322
12.3.2	使用 select 模块进行异步 I/O .....	324
任务 12-3	简单的聊天室 .....	327
12.4	FTP 文件传输 .....	334
任务 12-4	FTP 文件批量下载 .....	335
12.5	SMTP 发送邮件 .....	338
任务 12-5	复杂内容邮件发送 .....	339
12.6	习题 .....	342
<b>单元 13</b>	<b>Web 编程 .....</b>	<b>343</b>
13.1	Web 客户端访问 .....	343
13.1.1	Web 访问模块简介 .....	343
13.1.2	访问普通 Web 页面 .....	348
13.1.3	提交表单数据 .....	349
任务 13-1	网页爬虫 .....	350
13.2	Web 开发 .....	352
13.2.1	Tornado 服务器 .....	353
任务 13-2	表单提交 .....	356
13.2.2	SQLAlchemy 模块 .....	359
任务 13-3	一个简单的 MVC 网站 .....	360
13.3	习题 .....	369
<b>单元 14</b>	<b>Python 工程应用 .....</b>	<b>370</b>
14.1	NumPy 模块 .....	370
14.2	SciPy 模块 .....	371

14.2.1	SciPy 数值计算 .....	372
任务 14-1	最小二乘拟合 .....	374
14.2.2	SciPy 矩阵运算 .....	375
14.2.3	SciPy 图像处理 .....	376
任务 14-2	图像去噪 .....	379
14.3	Matplotlib 模块 .....	381
14.3.1	快速绘图 .....	382
14.3.2	绘制子图 .....	383
14.3.3	绘制各类图形 .....	387
14.3.4	使用 Latex .....	390
14.4	习题 .....	390
参考文献	.....	391

# 初级篇

搭建环境和运行 Python 应用程序

## Python 基础语法





## 搭建环境和运行 Python 应用程序

Python 是一种跨平台的面向对象的程序设计语言,具有简单性、易学性、开源性、可移植性、可扩展性和丰富类库支持等特点,是目前非常流行的程序设计语言之一。Python 广泛应用于窗口界面程序开发、网络程序开发、嵌入式程序开发和机器学习开发等领域。Eclipse+Pydev 是较适合初学者学习和使用的开发工具之一。

### 1.1 认识 Python

#### 1.1.1 Python 的由来

1989 年,CWI(阿姆斯特丹国家数学和计算机科学研究所)的研究员 Guido van Rossum 需要一种高级脚本语言来为 Amoeba 分布式操作系统执行管理任务。他决定开发一个新的继承自高级数学语言 ABC(All Basic Code)的解释型脚本语言,并命名为 Python。该名字来源于他喜爱的 BBC 电视剧 *Monty Python's Flying Circus*。ABC 也是由 Guido 参加设计的一种为非专业程序员设计的教学语言,非常优美和强大。但是 ABC 语言没有获得成功,Guido 认为是由于非开放式造成的。他从 ABC 中汲取了大量语法,从系统编程语言 Modula-3 借鉴了错误处理机制,并且完美结合了 UNIX Shell 和 C 以及其他一些语言,设计出一种开源的面向对象的脚本语言——Python。

Python 的设计哲学是“优雅”“明确”“简单”。Python 开发者的哲学是“用一种方法,最好是只用一种方法来做一件事”。在设计 Python 语言时,如果面临多种选择,Python 开发者一般会拒绝复杂的语法,而选择明确没有或者很少有歧义的语法。这些准则称为“Python 格言”。

由于 Python 语言的简洁性、易读性以及可扩展性,在国外用 Python 做科学计算的研究机构日益增多,一些知名大学采用 Python 来教授程序设计课程。例如,美国卡耐基·梅隆大学的编程基础、麻省理工学院的计算机科学及编程导论就使用 Python 语言讲授。另外,众多开源的科学计算软件包都提供 Python 的调用接口,例如著名的计算机视觉库 OpenCV、三维可视化库 VTK、医学图像处理库 ITK 等。Python 专用的科学计算扩展库就更多了,例如下面 3 个十分经典的科学计算扩展库: NumPy、SciPy 和 Matplotlib,为 Python 提供了快速数组处理、数值运算以及绘图功能。因此,Python 语言及其众多的扩展库所构成的开发环境十分适合工程技术人员及科研人员处理实验数据、制作图表,甚至开发科学计算应用程序。



### 1.1.2 Python 的特色

Python 是一种应用较为广泛的计算机语言,具有如下特点。

(1) 简单: Python 是一种代表简单主义思想的语言。一个好的 Python 程序阅读起来就感觉像是在读英语文章一样。Python 的这种伪代码本质,使得人们能够专注于解决问题,而不是去弄清语言本身。

(2) 易学: Python 的语法很简单,并且在使用变量之前不需要声明变量的类型。使用 Python 编程,不必像 C 语言那样关注内存空间的使用,它可以自动地进行内存分配和回收。另外,Python 提供了功能强大的内置对象和方法。

(3) 免费、开源: Python 是 FLOSS(自由/开放源码软件)之一。用户可以自由地发布该软件的拷贝,阅读源代码,进行修改,用在新的自由软件中等。用户在使用过程中不需要支付任何费用,也不存在版权问题。

(4) 可移植性: Python 被移植在许多平台上(经过改动,使它能够在不同平台上)。Python 程序无须修改,就可以在许多平台上运行。

(5) 解释性: Python 语言写的程序不需要编译成二进制代码。Python 程序运行时,首先由 Python 解释器把源代码转换为称为字节码的中间代码形式,然后翻译成计算机使用的机器语言并运行。

(6) 面向对象: Python 既支持面向过程的编程,也支持面向对象的编程。在面向过程编程中,Python 程序由模块或函数来构建。在面向对象编程中,Python 程序通过数据和功能组合形成的类来构建。与 C++ 和 Java 相比,Python 的面向对象编程更简单。

(7) 可扩展性: Python 的扩展接口可以把 Python 代码嵌入 C 或 C++ 程序,也可以在 Python 程序中调用使用 C 或 C++ 编写的代码。

(8) 丰富的库: Python 标准库很庞大,并且可以加载数量庞大的第三方库。因此,可以快速构建相关应用程序,如网络应用程序、数据库应用程序、多线程应用程序、GUI 界面程序等。

### 1.1.3 Python 的开发工具

IDE(Integration Development Environment)是指集成开发环境,以代码编辑器为核心,包括一系列周边组件和附属功能的软件开发工具。一个优秀的 IDE,除了提供普通文本编辑之外,还需要提供针对特定语言的各种快捷编辑功能,让程序员尽可能快捷、舒适、清晰地浏览、输入、修改代码。对于 IDE 来说,语法着色、错误提示、代码折叠、代码完成、代码块定位、重构,以及与调试器、版本控制系统(VCS)的集成等都是重要的功能。通过插件、扩展系统为代表的可定制框架,是 IDE 的流行趋势。

下面介绍几个流行的 IDE 工具。

IDLE 是 Python 标准发行版内置的一个简单、小巧的 IDE,包括交互式命令行、编辑器、调试器等基本组件,足以应付大多数简单应用。IDLE 是用纯 Python 基于 Tkinter 编写的,其最初的作者正是 Python 之父 Guido van Rossum 本人。

PythonWin 是 Python Win32 Extensions(半官方性质的 Python for Win32 增强包)的一部分,也包含在 ActivePython 的 Windows 发行版中,但它只能用于 Win32 平台。