

Python

从入门到精通

36小时视频讲解

明日科技 编著

✓ 100个精彩实例

✓ 235集教学微视频

✓ Python背记手册

✓ PPT电子课件

循序渐进，实战讲述

基础知识 ⇨ 核心技术 ⇨ 高级应用 ⇨ 项目实战

100个精彩实例，7大核心应用，1个项目案例

趣味解读，易学易用

Python 核心知识点 + 趣味案例 + 项目案例 + 背记手册

视频 + 语音，尽享编程乐趣

235集高清微视频，可看可听，随时随地扫码学

在线解答，高效学习



清华大学出版社

软件

Python 从入门到精通

明日科技 编著

清华大学出版社

北京

内 容 简 介

《Python 从入门到精通》从初学者角度出发，通过通俗易懂的语言、丰富多彩的实例，详细介绍了使用 Python 进行程序开发应该掌握的各方面技术。全书共分 22 章，包括初识 Python、Python 语言基础、运算符与表达式、流程控制语句、列表与元组、字典与集合、字符串、Python 中使用正则表达式、函数、面向对象程序设计、模块、异常处理及程序调试、文件及目录操作、操作数据库、GUI 界面编程、Pygame 游戏编程、网络爬虫开发、使用进程和线程、网络编程、Web 编程、Flask 框架、e 起去旅行网站等。所有知识都结合具体实例进行介绍，涉及的程序代码都给出了详细的注释，读者可轻松领会 Python 程序开发的精髓，快速提升开发技能。除此之外，本书还附配了 235 集高清教学微视频及 PPT 电子教案。

本书可作为软件开发入门者的学习用书，也可作为高等院校相关专业的教学参考用书，还可供开发人员查阅、参考使用。

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 从入门到精通 / 明日科技编著. —北京：清华大学出版社，2018
(软件开发视频大讲堂)
ISBN 978-7-302-50388-0

I. ①P… II. ①明… 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 (2018) 第 122956 号

责任编辑：贾小红
封面设计：梁 英
版式设计：楠竹文化
责任校对：赵丽杰
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
质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn

印 装 者：三河市龙大印装有限公司

经 销：全国新华书店

开 本：203mm×260mm 印 张：32.75 字 数：866 千字
版 次：2018 年 10 月第 1 版 印 次：2018 年 10 月第 1 次印刷
定 价：79.80 元

产品编号：078932-01

前言

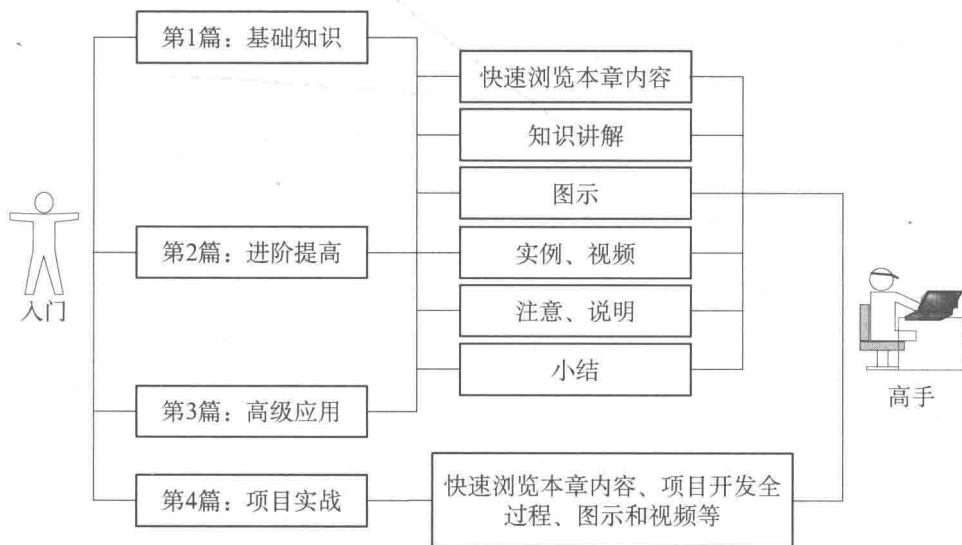
Preface

Python 人称“胶水”语言，能够把用其他语言制作的各种模块（尤其是 C/C++）很轻松地联结在一起。它是 1989 年由荷兰人 Guido van Rossum 发明的一种面向对象的解释型高级编程语言。由于 Python 语言简洁、易读，非常适合编程入门，现在很多学校都开设了这门课程，甚至有些小学也开设了 Python 课程。连小学生都能学会的语言，您还在等什么呢？快快加入 Python 开发者的阵营吧！

目前，关于 Python 的书籍有很多，但是真正适合初学者学习的书籍并不是很多，尤其是以 Python 3 为主体的书籍就更少了。本书从初学者的角度出发，循序渐进地讲解使用 Python 开发应用项目和游戏时应该掌握的各项技术。

本书内容

本书提供了从入门到编程高手所必备的各类知识，共分 4 篇，大体结构如下图所示。



第 1 篇：基础知识。本篇包括 Python 简介、搭建 Python 开发环境、Python 开发工具、Python 语法特点、Python 中的变量、基本数据类型、基本输入和输出、运算符与表达式、流程控制语句、列表与元组、字典与集合以及字符串等语言基础方面的知识。介绍时结合大量的图示、举例、视频，使读者能快速掌握 Python 语言，并为以后编程奠定坚实的基础。

第 2 篇：进阶提高。本篇包括 Python 中使用正则表达式、函数、面向对象程序设计、模块、异常处理及程序调试、文件及目录操作、操作数据库等内容。学习完本篇，读者可以掌握更深一层的 Python 开发技术。

第3篇：高级应用。本篇包括 GUI 界面编程、Pygame 游戏编程、网络爬虫开发、使用进程和线程、网络编程、Web 编程、Flask 框架等内容。学习完本篇，读者将能够开发 GUI 界面程序、简单的游戏、网络爬虫、网络及 Web 程序等。

第4篇：项目实战。本篇通过一个完整的 Web 项目——e 起去旅行网站，运用软件工程的设计思想，引导读者学习如何进行软件项目的实践开发。书中按照“系统功能设计→数据库设计→前台模块设计→后台模块设计”的流程进行介绍，带领读者亲身体验使用 Flask 框架开发 Web 项目的全过程。

本书特点

- **由浅入深，循序渐进。**本书以初、中级程序员为对象，先从 Python 语言基础学起，然后学习 Python 的进阶与提高技术，接下来再学习 Python 的高级应用，最后学习开发一个完整的 Web 项目。讲解过程中步骤详尽，版式新颖，在操作的内容图片上以①②③……编号+内容的方式进行标注，让读者在阅读中一目了然，从而快速把握书中内容。
- **语音视频，讲解详尽。**对于初学者来说，视频讲解是最好的导师，它能够引导初学者快速入门，使初学者感受到编程的快乐和成就感，进一步增强学习的信心。鉴于此，本书为大部分章节都配备了视频讲解，使用手机扫描正文小节标题一侧的二维码，即可在线学习程序开发的全过程。
- **实例典型，轻松易学。**通过实例学习是最好的学习方式，本书通过“一个知识点、一个例子、一个结果、一段评析、一个综合应用”的模式，透彻详尽地讲述了实际开发中所需的各类知识。另外，为了便于读者阅读程序代码，快速学习编程技能，书中几乎每行代码都提供了注释。
- **精彩栏目，贴心提醒。**本书根据需要在各章使用了很多“注意”“说明”“常见错误”等小栏目，读者可以在学习过程中轻松理解相关知识点及概念，快速掌握相应技术的应用技巧。

读者对象

- | | |
|---|--|
| <input checked="" type="checkbox"/> 初学编程的自学者 | <input checked="" type="checkbox"/> 编程爱好者 |
| <input checked="" type="checkbox"/> 大中专院校的老师 and 学生 | <input checked="" type="checkbox"/> 相关培训机构的老师和学员 |
| <input checked="" type="checkbox"/> 做毕业设计的学生 | <input checked="" type="checkbox"/> 初、中级程序开发人员 |
| <input checked="" type="checkbox"/> 程序测试及维护人员 | <input checked="" type="checkbox"/> 参加实习的“菜鸟”程序员 |

读者服务

学习本书时，请先扫描封底的权限二维码（需要刮开涂层）获取学习权限，然后即可免费学习书中的所有线上线下资源。本书所附赠的学习资源包，读者可登录清华大学出版社网站（www.tup.com.cn），

在对应图书页面下获取其下载方式。也可扫描图书封底的“文泉云盘”二维码，获取其下载方式。

为了方便读者，本书提供了学习答疑网站（www.mingrisoft.com），有关本书的问题，读者均可在网站上留言，我们力求在 24 小时内回复（节假日除外）。

致读者

本书由明日科技 Python 程序开发团队组织编写，主要编写人员有王国辉、冯春龙、李磊、赛奎春、王小科、申小琦、辛洪郁、张鑫、杨丽、周佳星、赵宁、李菁菁、白宏健、贾景波、申野、庞凤、张云凯、梁英、张宝华、杨丽、杨柳、宋万勇、刘杰、隋妍妍、葛忠月、高春艳、朱艳红、宋禹蒙、岳彩龙、李春林等。在编写本书的过程中，我们以科学、严谨的态度，力求精益求精，但错误、疏漏之处在所难免，敬请广大读者批评指正。我们的服务邮箱是 mingrisoft@mingrisoft.com。读者在阅读本书时，如果发现错误或遇到问题，可以发送电子邮件及时与我们联系，我们会尽快给予答复。

感谢您购买本书，希望本书能成为您编程路上的领航者。





“零门槛”编程，一切皆有可能。祝读书快乐！




编 者

目 录





Contents




第 1 篇 基础知识

第 1 章 初识 Python	2	2.3.1 数字	35
 视频讲解: 68 分钟		2.3.2 字符串	38
1.1 Python 简介	3	2.3.3 布尔类型	40
1.1.1 什么是 Python	3	2.3.4 数据类型转换	40
1.1.2 Python 的版本	4	2.4 基本输入和输出	42
1.1.3 Python 都能做什么	5	2.4.1 使用 input()函数输入	42
1.2 搭建 Python 开发环境	7	2.4.2 使用 print()函数输出	43
1.2.1 开发环境概述	7	2.5 小结	44
1.2.2 安装 Python	7	第 3 章 运算符与表达式	45
1.2.3 第一个 Python 程序	12	 视频讲解: 64 分钟	
1.2.4 运行已经编写好的.py 文件	14	3.1 运算符	46
1.3 Python 开发工具	15	3.1.1 算术运算符	46
1.3.1 使用自带的 IDLE	15	3.1.2 赋值运算符	47
1.3.2 常用的第三方开发工具	18	3.1.3 比较(关系)运算符	48
1.4 小结	20	3.1.4 逻辑运算符	49
第 2 章 Python 语言基础	21	3.1.5 位运算符	51
 视频讲解: 135 分钟		3.2 运算符的优先级	54
2.1 Python 语法特点	22	3.3 条件表达式	55
2.1.1 注释规则	22	3.4 小结	56
2.1.2 代码缩进	26	第 4 章 流程控制语句	57
2.1.3 编码规范	28	 视频讲解: 106 分钟	
2.2 Python 中的变量	30	4.1 程序结构	58
2.2.1 保留字与标识符	30	4.2 选择语句	59
2.2.2 理解 Python 中的变量	33	4.2.1 最简单的 if 语句	59
2.2.3 定义变量	34	4.2.2 if...else 语句	61
2.3 基本数据类型	35		

4.2.3	if...elif...else 语句	64	5.3.3	修改元组	108
4.2.4	if 语句的嵌套	66	5.3.4	元组推导式	109
4.3	循环语句	68	5.3.5	元组与列表的区别	111
4.3.1	while 循环	68	5.4	小结	111
4.3.2	for 循环	69	第 6 章	字典与集合	112
4.3.3	循环嵌套	72	 视频讲解: 86 分钟		
4.4	break、continue 和 pass 语句	74	6.1	字典	113
4.4.1	break 语句	74	6.1.1	字典的创建和删除	113
4.4.2	continue 语句	76	6.1.2	访问字典	116
4.4.3	pass 语句	77	6.1.3	遍历字典	119
4.5	小结	78	6.1.4	添加、修改和删除字典元素	120
第 5 章	列表与元组	79	6.1.5	字典推导式	121
 视频讲解: 202 分钟			6.2	集合	122
5.1	序列概述	80	6.2.1	创建集合	122
5.1.1	索引	80	6.2.2	向集合中添加和删除元素	124
5.1.2	切片	81	6.2.3	集合的交集、并集和差集运算	126
5.1.3	序列相加	82	6.3	小结	127
5.1.4	乘法	82	第 7 章	字符串	128
5.1.5	检查某个元素是否是序列的 成员 (元素)	83	 视频讲解: 118 分钟		
5.1.6	计算序列的长度、最大值和最小值	83	7.1	字符串编码转换	129
5.2	列表	84	7.1.1	使用 encode() 方法编码	129
5.2.1	列表的创建和删除	85	7.1.2	使用 decode() 方法解码	130
5.2.2	访问列表元素	87	7.2	字符串常用操作	131
5.2.3	遍历列表	88	7.2.1	拼接字符串	131
5.2.4	添加、修改和删除列表元素	90	7.2.2	计算字符串的长度	132
5.2.5	对列表进行统计计算	94	7.2.3	截取字符串	133
5.2.6	对列表进行排序	96	7.2.4	分割、合并字符串	135
5.2.7	列表推导式	98	7.2.5	检索字符串	137
5.2.8	二维列表	99	7.2.6	字母的大小写转换	141
5.3	元组	103	7.2.7	去除字符串中的空格和 特殊字符	142
5.3.1	元组的创建和删除	103	7.2.8	格式化字符串	145
5.3.2	访问元组元素	106	7.3	小结	148





第 2 篇 进阶提高

第 8 章 Python 中使用正则表达式	150	9.4 变量的作用域	179
 视频讲解: 45 分钟		9.4.1 局部变量	180
8.1 正则表达式语法	151	9.4.2 全局变量	180
8.1.1 行定位符	151	9.5 匿名函数	182
8.1.2 元字符	151	9.6 小结	184
8.1.3 重复	152	第 10 章 面向对象程序设计	185
8.1.4 字符类	152	 视频讲解: 110 分钟	
8.1.5 排除字符	153	10.1 面向对象概述	186
8.1.6 选择字符	153	10.1.1 对象	186
8.1.7 转义字符	153	10.1.2 类	187
8.1.8 分组	154	10.1.3 面向对象程序设计的特点	187
8.1.9 在 Python 中使用正则 表达式语法	154	10.2 类的定义和使用	189
8.2 使用 re 模块实现正则表达式操作	155	10.2.1 定义类	189
8.2.1 匹配字符串	155	10.2.2 创建类的实例	190
8.2.2 替换字符串	160	10.2.3 创建 <code>__init__()</code> 方法	190
8.2.3 使用正则表达式分割字符串	161	10.2.4 创建类的成员并访问	192
8.3 小结	162	10.2.5 访问限制	197
第 9 章 函数	163	10.3 属性	198
 视频讲解: 118 分钟		10.3.1 创建用于计算的属性	198
9.1 函数的创建和调用	164	10.3.2 为属性添加安全保护机制	199
9.1.1 创建一个函数	164	10.4 继承	201
9.1.2 调用函数	166	10.4.1 继承的基本语法	201
9.2 参数传递	167	10.4.2 方法重写	203
9.2.1 了解形式参数和实际参数	167	10.4.3 派生类中调用基类的 <code>__init__()</code> 方法	204
9.2.2 位置参数	170	10.5 小结	206
9.2.3 关键字参数	172	第 11 章 模块	207
9.2.4 为参数设置默认值	172	 视频讲解: 105 分钟	
9.2.5 可变参数	174	11.1 模块概述	208
9.3 返回值	178		

11.2 自定义模块	208	13.1.3 打开文件时使用 with 语句	252
11.2.1 创建模块	208	13.1.4 写入文件内容	252
11.2.2 使用 import 语句导入模块	209	13.1.5 读取文件	255
11.2.3 使用 from...import 语句 导入模块	210	13.2 目录操作	260
11.2.4 模块搜索目录	213	13.2.1 os 和 os.path 模块	260
11.3 Python 中的包	216	13.2.2 路径	262
11.3.1 Python 程序的包结构	216	13.2.3 判断目录是否存在	264
11.3.2 创建和使用包	217	13.2.4 创建目录	265
11.3.3 以主程序的形式执行	221	13.2.5 删除目录	267
11.4 引用其他模块	224	13.2.6 遍历目录	268
11.4.1 导入和使用标准模块	224	13.3 高级文件操作	270
11.4.2 第三方模块的下载与安装	226	13.3.1 删除文件	271
11.5 小结	228	13.3.2 重命名文件和目录	272
第 12 章 异常处理及程序调试	229	13.3.3 获取文件基本信息	273
 视频讲解: 52 分钟		13.4 小结	275
12.1 异常概述	230	第 14 章 操作数据库	276
12.2 异常处理语句	232	 视频讲解: 74 分钟	
12.2.1 try...except 语句	232	14.1 数据库编程接口	277
12.2.2 try...except...else 语句	235	14.1.1 连接对象	277
12.2.3 try...except...finally 语句	236	14.1.2 游标对象	278
12.2.4 使用 raise 语句抛出异常	238	14.2 使用 SQLite	279
12.3 程序调试	240	14.2.1 创建数据库文件	279
12.3.1 使用自带的 IDLE 进行程序调试	240	14.2.2 操作 SQLite	280
12.3.2 使用 assert 语句调试程序	244	14.3 使用 MySQL	284
12.4 小结	246	14.3.1 下载安装 MySQL	284
第 13 章 文件及目录操作	247	14.3.2 安装 PyMySQL	290
 视频讲解: 140 分钟		14.3.3 连接数据库	290
13.1 基本文件操作	248	14.3.4 创建数据表	291
13.1.1 创建和打开文件	248	14.3.5 操作 MySQL 数据表	292
13.1.2 关闭文件	251	14.4 小结	294

第 3 篇 高级应用

第 15 章 GUI 界面编程	296	16.3.2 游戏分析	325
📺 视频讲解: 82 分钟		16.3.3 搭建主框架	325
15.1 初识 GUI	297	16.3.4 创建小鸟类	327
15.1.1 什么是 GUI	297	16.3.5 创建管道类	329
15.1.2 常用的 GUI 框架	297	16.3.6 计算得分	331
15.1.3 安装 wxPython	298	16.3.7 碰撞检测	333
15.2 创建应用程序	299	16.4 小结	335
15.2.1 创建一个 wx.App 的子类	299	第 17 章 网络爬虫开发	336
15.2.2 直接使用 wx.App	300	📺 视频讲解: 163 分钟	
15.2.3 使用 wx.Frame 框架	301	17.1 初识网络爬虫	337
15.3 常用控件	302	17.1.1 网络爬虫概述	337
15.3.1 StaticText 文本类	302	17.1.2 网络爬虫的分类	337
15.3.2 TextCtrl 输入文本类	305	17.1.3 网络爬虫的基本原理	338
15.3.3 Button 按钮类	306	17.2 网络爬虫的常用技术	339
15.4 BoxSizer 布局	308	17.2.1 Python 的网络请求	339
15.4.1 什么是 BoxSizer	308	17.2.2 请求 headers 处理	342
15.4.2 使用 BoxSizer 布局	308	17.2.3 网络超时	343
15.5 事件处理	312	17.2.4 代理服务	344
15.5.1 什么是事件	312	17.2.5 HTML 解析之 BeautifulSoup	345
15.5.2 绑定事件	312	17.3 网络爬虫开发常用框架	348
15.6 小结	314	17.3.1 Scrapy 爬虫框架	348
第 16 章 Pygame 游戏编程	315	17.3.2 Crawley 爬虫框架	349
📺 视频讲解: 84 分钟		17.3.3 PySpider 爬虫框架	349
16.1 初识 Pygame	316	17.4 实战项目: 快手爬票	350
16.1.1 安装 Pygame	316	17.4.1 快手爬票概述	350
16.1.2 Pygame 常用模块	317	17.4.2 搭建 QT 环境	351
16.2 Pygame 基本使用	319	17.4.3 主窗体设计	355
16.3 开发 Flappy Bird 游戏	324	17.4.4 分析网页请求参数	364
16.3.1 游戏简介	324	17.4.5 下载站名文件	366

17.4.6	车票信息的请求与显示	369	19.3	UDP 编程	411
17.5	小结	376	19.3.1	创建 UDP 服务器	412
第 18 章	使用进程和线程	377	19.3.2	创建 UDP 客户端	412
	 视频讲解：95 分钟		19.3.3	执行 UDP 服务器和客户端	413
18.1	什么是进程	378	19.4	小结	414
18.2	创建进程的常用方式	379	第 20 章	Web 编程	415
18.2.1	使用 multiprocessing 模块创建进程	379		 视频讲解：93 分钟	
18.2.2	使用 Process 子类创建进程	382	20.1	Web 基础	416
18.2.3	使用进程池 Pool 创建进程	384	20.1.1	HTTP 协议	416
18.3	进程间通信	386	20.1.2	Web 服务器	416
18.3.1	队列简介	388	20.1.3	前端基础	419
18.3.2	多进程队列的使用	388	20.1.4	静态服务器	421
18.3.3	使用队列在进程间通信	390	20.2	WSGI 接口	426
18.4	什么是线程	391	20.2.1	CGI 简介	426
18.5	创建线程	391	20.2.2	WSGI 简介	427
18.5.1	使用 threading 模块创建线程	392	20.2.3	定义 WSGI 接口	428
18.5.2	使用 Thread 子类创建线程	393	20.2.4	运行 WSGI 服务	428
18.6	线程间通信	394	20.3	Web 框架	431
18.6.1	什么是互斥锁	395	20.3.1	什么是 Web 框架	431
18.6.2	使用互斥锁	396	20.3.2	Python 中常用的 Web 框架	431
18.6.3	使用队列在线程间通信	398	20.4	小结	432
18.7	小结	400	第 21 章	Flask 框架	433
第 19 章	网络编程	401		 视频讲解：75 分钟	
	 视频讲解：60 分钟		21.1	Flask 简介	434
19.1	网络基础	402	21.1.1	安装虚拟环境	434
19.1.1	为什么要使用通信协议	402	21.1.2	安装 Flask	435
19.1.2	TCP/IP 简介	402	21.1.3	第一个 Flask 程序	436
19.1.3	UDP 简介	404	21.2	Flask 基础	437
19.1.4	Socket 简介	405	21.2.1	开启调试模式	437
19.2	TCP 编程	406	21.2.2	路由	438
19.2.1	创建 TCP 服务器	406	21.2.3	静态文件	440
19.2.2	创建 TCP 客户端	408	21.2.4	蓝图	441
19.2.3	执行 TCP 服务器和客户端	409	21.3	模板	441

21.3.1 渲染模板	441	21.4.1 CSRF 保护和验证	446
21.3.2 变量	443	21.4.2 表单类	447
21.3.3 控制结构	444	21.4.3 把表单渲染成 HTML	449
21.4 Web 表单	446	21.5 小结	452

第 4 篇 项目实战

第 22 章 e 起去旅行网站	454	22.5.1 推荐景区功能实现	472
 视频讲解: 143 分钟		22.5.2 推荐地区功能实现	474
22.1 系统功能设计	455	22.5.3 搜索景区功能实现	475
22.1.1 系统功能结构	455	22.6 景区模块设计	478
22.1.2 系统业务流程	455	22.6.1 查看景区功能实现	478
22.2 系统开发必备	456	22.6.2 查看游记功能实现	481
22.2.1 系统开发环境	456	22.6.3 收藏景区功能实现	482
22.2.2 文件夹组织结构	456	22.6.4 查看收藏景区功能实现	485
22.3 数据库设计	458	22.7 关于我们模块设计	488
22.3.1 数据库概要说明	458	22.8 后台模块设计	492
22.3.2 数据表模型	458	22.8.1 管理员登录功能实现	492
22.3.3 数据表关系	460	22.8.2 景区管理功能实现	493
22.4 前台用户模块设计	460	22.8.3 地区管理功能实现	505
22.4.1 会员注册功能实现	460	22.8.4 游记管理功能实现	505
22.4.2 会员登录功能实现	468	22.8.5 会员管理功能实现	506
22.4.3 会员退出功能实现	470	22.8.6 日志管理功能实现	507
22.5 前台首页模块设计	470	22.9 小结	508

第 1 篇

基础知识

- » 第 1 章 初识 Python
- » 第 2 章 Python 语言基础
- » 第 3 章 运算符与表达式
- » 第 4 章 流程控制语句
- » 第 5 章 列表与元组
- » 第 6 章 字典与集合
- » 第 7 章 字符串

本篇通过对走进 Python、Python 语言基础、运算符与表达式、流程控制语句、列表与元组、字典与集合、字符串等内容的介绍，并结合大量的图示、举例、视频等使您快速掌握 Python 语言，为以后编程奠定坚实的基础。

第 1 章

初识 Python

( 视频讲解：68 分钟)

Python 是一种跨平台的、开源的、免费的、解释型的高级编程语言。近几年发展势头迅猛，在 2018 年 3 月的 TIOBE 编程语言排行榜中已经晋升到第 4 名，而在 IEEE Spectrum 发布的 2017 年度编程语言排行榜中，Python 位居第一。另外，Python 的应用领域非常广泛，如 Web 编程、图形处理、黑客编程、大数据处理、网络爬虫和科学计算等，Python 都可以实现。

作为 Python 开发的起步，本章将先对学习 Python 需要了解的一些基础内容进行简要介绍，然后重点介绍如何搭建 Python 开发环境，最后介绍常见的几种 Python 的开发工具。

通过阅读本章，您可以：

- ▶▶ 了解什么是 Python 以及 Python 的版本和 Python 能做什么
- ▶▶ 掌握如何搭建 Python 的开发环境
- ▶▶ 掌握如何在命令行窗口中编写 Python 程序
- ▶▶ 掌握通过 IDLE 编写 Python 程序
- ▶▶ 掌握如何运行已经编写好的.py 文件
- ▶▶ 掌握 Python 自带的 IDLE 的基本使用方法
- ▶▶ 了解 Python 常用的第三方开发工具

1.1 Python 简介



1.1.1 什么是 Python

Python（发音[ˈpaɪθən]）本义是指“蟒蛇”（这里需要说明的是，Python 并不是以蟒蛇命名，而是以电视节目 Monty Python's Flying Circus 来命名的），标志如图 1.1 所示。它是 1989 年由荷兰人 Guido van Rossum 发明的一种面向对象的解释型高级编程语言。它的设计哲学为优雅、明确、简单。实际上，Python 也是按照这个理念做的，以至于现在网络上流传着“人生苦短，我用 Python”的说法。可见 Python 有着简单、开发速度快、节省时间和精力等特点。

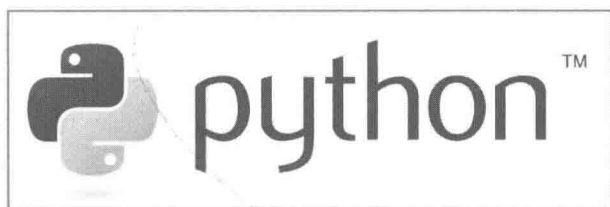


图 1.1 Python 的标志

Python 本身并非所有的特性和功能都集成到语言核心，而是被设计为可扩充的。它具有丰富和强大的库，能够把用其他语言（尤其是 C/C++）制作的各种模块很轻松地联结在一起。为此，Python 常被称为“胶水”语言。*

在 1991 年 Python 的第一个公开发行人问世之后，Python 的发展并不突出。自从 2004 年以后，Python 的使用率呈线性增长。在 2010 年时，Python 赢得 TIOBE 2010 年度语言大奖。在 2017 年，IEEE Spectrum 发布的 2017 年度编程语言排行榜中，Python 位居第一，如图 1.2 所示。

Language Rank	Types	Spectrum Ranking
1. Python	🌐 🖥️	100.0
2. C	📱 🖥️ 🐛	99.7
3. Java	🌐 📱 🖥️	99.5
4. C++	📱 🖥️ 🐛	97.1
5. C#	🌐 📱 🖥️	87.7
6. R	🖥️	87.7
7. JavaScript	🌐 📱	85.6
8. PHP	🌐	81.2
9. Go	🌐 🖥️	75.1
10. Swift	📱 🖥️	73.7

图 1.2 IEEE Spectrum 发布的 2017 年度编程语言排行榜前 10 名

1.1.2 Python 的版本

Python 自发布以来,主要经历了 3 个版本的变化。分别是 1994 年发布的 Python 1.0 版本(已过时),2000 年发布的 Python 2.0 版本(现在已经更新到 2.7.x)和 2008 年发布的 3.0 版本(现在已经更新到 3.6.x)。

1. Python 2.x 和 3.x 的区别

Python 在版本升级时,并不是向下兼容的。在 Python 的官方网站中同时发布了两个不同系列的版本,分别是 Python 2.x 版本和 Python 3.x 版本,它们之间在基本语法上主要存在以下区别。

- ☑ 在 Python 2.x 中, print 语句被 Python 3.x 中的 print()函数所代替。
- ☑ 在 Python 3.x 中,整数之间的相除(采用除法运算符“/”实现),结果是浮点数,而在 Python 2.x 中结果是整数。
- ☑ Python 3.x 源码文件默认使用 UTF-8 编码,所以支持直接写入的中文,而 Python 2.x 默认编码是 ASCII,直接写入中文会被转换为 ANSI 编码。所以在 Python 2.x 中需要进行相应的转换。
- ☑ 在 Python 3.x 中将 range()与 xrange()函数整合为一个 range()函数,所以在 Python 3.x 中不存在 xrange()函数,而在 Python 2.x 中这两个函数是并存的。



说明

除了以上列出的几点主要区别,Python 3.x 和 Python 2.x 还有一些其他的区别,这里将不再列举,在后面的章节中涉及时再进行详细介绍。

2. 初学者应该选择哪个版本

目前,根据 Semaphore 社区的调查结果,使用 Python 2.x 的开发者占 63.7%,而使用 Python 3.x 的用户占 36.3%,由此可见,使用 Python 2.x 的还是占多数。并且 Python 的作者曾于 2014 年宣布 Python 2.7 支持时间延长到 2020 年。那么作为初学者应该选择什么版本呢?

答:笔者建议初学者应该选择 Python 3.x 版本。理由主要有以下几点。

- ☑ 使用 Python 3.x 毕竟是大势所趋

虽然目前使用 Python 2.x 的开发者居多,但是使用 Python 3.x 的开发者更愿意进行版本更新,并且使用 Python 3.x 版本的开发者正在迅速扩展,如图 1.3 所示。

- ☑ Python 3.x 较 Python 2.x 有很大改进

Python 3.x 对 Python 2.x 的标准库进行了一定程度的重新拆分和整合,使得它比 Python 2.x 更容易理解,特别是在字符编码方面。Python 2.x 中对于中文字符串支持不好,需要编写单独的代码对中文进行处理,否则不能正确显示中文。但是在 Python 3.x 中已经成功地解决了这一难题。

- ☑ Python 3.x 和 Python 2.x 思想基本是共通的

Python 3.x 和 Python 2.x 思想基本是共通的,只有少量的语法差别。学会了 Python 3.x,只要稍微花一点时间学习 Python 2.x 的语法,两种语言就都学会了。