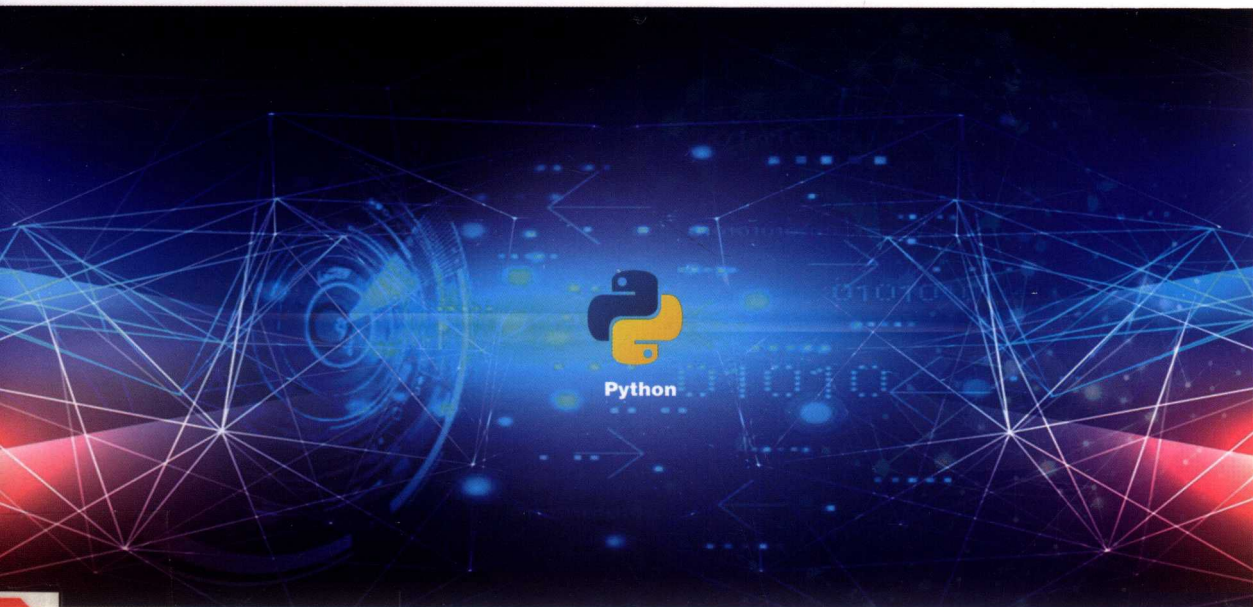



高级大数据人才培养丛书

# Python程序设计

BIG DATA

刘鹏 张燕 ○ 丛书主编 张雪萍 ○ 主编 唐万梅 景雪琴 ○ 副主编



 中国工信出版集团

 电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY  
<http://www.phei.com.cn>

高级大数据人才培养丛书

# Python 程序设计

丛书主编：刘 鹏 张 燕

主 编：张雪萍

副主编：唐万梅 景雪琴

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书是中国信息协会大数据分会副会长刘鹏教授组织编撰的“高级大数据人才培养丛书”之一。本书是一本全面的、从入门到实践的 Python 编程教程，从带领读者快速掌握基本的 Python 编程知识开始，循序渐进、层层深入地引导读者利用新学到的知识开发功能丰富的项目。本书首先介绍了 Python 基础；接着介绍了数据类型、文件、程序调试、面向对象程序设计、连接数据源等基本知识；然后结合网络爬虫、数据挖掘、自然语言处理、数据可视化、Web 和移动应用等工具，以案例为依托进行项目实战；最后介绍了国内各种云服务平台，以及如何运用 Python 实现访问。本书的全部实验均可在大数据实验平台 (<http://bd.cstor.cn>) 上远程开展，也可在高校部署的 BDRack 大数据实验一体机上本地开展。

本书系统全面、通俗易懂、结构合理，每章均有习题，可作为高等院校计算机及相关专业的本科和研究生教材，部分内容也可作为高职高专院校的教学内容。本书也适合作为编程人员的自学书籍。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有，侵权必究。

### 图书在版编目 (CIP) 数据

Python 程序设计 / 张雪萍主编. —北京: 电子工业出版社, 2019.4  
(高级大数据人才培养丛书 / 刘鹏, 张燕主编)

ISBN 978-7-121-36073-2

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 035760 号

策划编辑: 董亚峰

责任编辑: 米俊萍 特约编辑: 顾慧芳

印 刷: 三河市鑫金马印装有限公司

装 订: 三河市鑫金马印装有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

开 本: 787×1092 1/16 印张: 25 字数: 600 千字

版 次: 2019 年 4 月第 1 版

印 次: 2019 年 4 月第 1 次印刷

定 价: 88.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888, 88258888。

质量投诉请发邮件至 [zlts@phei.com.cn](mailto:zlts@phei.com.cn), 盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

本书咨询联系方式: [mijp@phei.com.cn](mailto:mijp@phei.com.cn)。

# 总序

---

短短几年间，大数据就以一日千里的发展速度，快速实现了从概念到落地，直接带动了相关产业井喷式发展。全球多家研究机构统计数据显示，大数据产业将迎来发展黄金期：IDC 预计，大数据和分析市场将从 2016 年的 1300 亿美元增长到 2020 年的 2030 亿美元以上；中国报告大厅发布的大数据行业报告数据也说明，自 2017 年起，我国大数据产业将迎来发展黄金期，未来 2~3 年的市场规模增长率将保持在 35% 左右。

数据采集、数据存储、数据挖掘、数据分析等大数据技术在越来越多的行业中得到应用，随之而来的就是大数据人才问题的凸显。麦肯锡预测，每年数据科学专业的应届毕业生将增加 7%，然而仅高质量项目对于专业数据科学家的需求每年就会增加 12%，完全供不应求。根据《人民日报》的报道，未来 3~5 年，中国需要 180 万数据人才，但目前只有约 30 万人，人才缺口近 150 万人。

以贵州大学为例，其首届大数据专业研究生就业率就达到 100%，可以说被“一抢而空”。急切的人才需求直接催热了大数据专业，教育部正式设立“数据科学与大数据技术”本科专业。目前已经有三批共计 283 所大学获批，包括北京大学、中南大学、对外经济贸易大学、中国人民大学、北京邮电大学、复旦大学等。

不过，就目前而言，在大数据人才培养和大数据课程建设方面，大部分高校仍然处于起步阶段，需要探索的问题还有很多。首先，大数据是个新生事物，懂大数据的老师少之又少，院校缺“人”；其次，尚未形成完善的大数据人才培养和课程体系，院校缺“机制”；再次，开展大数据实验需要为每位学生提供集群计算机，院校缺“机器”；最后，院校没有海量数据，开展大数据教学科研工作缺少“原材料”。

其实，早在网格计算和云计算兴起时，我国科技工作者就曾遇到过类似的挑战，我有幸参与了这些问题的解决过程。为了解决网格计算问题，我在清华大学读博期间，于 2001 年创办了中国网格信息中转站网站，每天花几个小时收集和分享有价值的资料给学术界，此后我也多次筹办和主持全国性的网格计算学术会议，进行信息传递与知识分享。2002 年，我与其他专家合作完成的《网格计算》教材也正式面世。

2008 年，当云计算开始萌芽之时，我创办了中国云计算网站（chinacloud.cn）（在各大搜索引擎“云计算”关键词搜索结果中排名前列），2010 年出版了《云计算》（第 1 版）、2011 年出版了《云计算》（第 2 版）、2015 年出版了《云计算》（第 3 版），每一版都花费了大量成本制作并免费分享对应的几十个教学 PPT。目前，这些 PPT 的下载总量达到了几百万次。同时，《云计算》一书也成为国内高校的首选教材。在 CNKI 公布的高被引图书

名单中,《云计算》(第1版)在自动化和计算机领域排名全国第一(统计了2010年后出版的所有图书)。除了资料分享,2010年我也在南京组织了全国高校云计算师资培训班,培养了国内第一批云计算老师,并通过与华为、中兴、360等知名企业合作,输出云计算技术,培养云计算研发人才。这些工作获得了大家的认可与好评,此后我接连担任了工信部云计算研究中心专家、中国云计算专家委员会云存储组组长等职务。

近几年,面对日益突出的大数据发展难题,我也在尝试使用此前类似的办法去应对这些挑战。为了解决大数据技术资料缺乏和交流不够通透的问题,我于2013年创办了中国大数据网站(thebigdata.cn),投入大量的人力进行日常维护;为了解决大数据师资匮乏的问题,我面向全国院校陆续举办多期大数据师资培训班。从2016年年底起,我在南京多次举办全国高校/高职/中职大数据免费培训班,基于《大数据》《大数据实验手册》及云创大数据提供的大数据实验平台,帮助到场老师们跑通了Hadoop、Spark等多个大数据实验,使他们跨过了“从理论到实践,从知道到用过”的门槛。2017年5月,我还举办了全国千所高校大数据师资免费讲习班,盛况空前。

其中,为了解决大数据实验难的问题而开发的大数据实验平台,正在为越来越多高校的教学科研带去方便:我带领云创大数据(www.cstor.cn,股票代码:835305)的科研人员,应用Docker容器技术,成功开发了BDRack大数据实验一体机,打破了虚拟化技术的性能瓶颈,可以为每位参加实验的人员虚拟出Hadoop集群、Spark集群、Storm集群等,自带实验所需数据,并准备了详细的实验手册(包含42个大数据实验)、PPT和实验过程视频,可以开展大数据管理、大数据挖掘等各类实验,并可以进行精确营销、信用分析等多种实战演练。目前,大数据实验平台已经在郑州大学、成都理工大学、金陵科技学院、天津农学院、西京学院、郑州升达经贸管理学院、信阳师范学院、镇江高等职业技术学校等多所院校成功应用,并广受校方好评。该平台也可在线使用(<http://bd.cstor.cn>),帮助师生通过自学,用一个月左右成为大数据实验动手的高手。此外,面对席卷而来的人工智能浪潮,我所在团队推出的AIRack人工智能实验平台、DeepRack深度学习一体机及dServer人工智能服务器等系列应用,一举解决了人工智能实验环境搭建困难、缺乏实验指导与实验数据等问题,目前已经在清华大学、南京大学、南京农业大学、西安科技大学等高校投入使用。

同时,为了解决缺乏权威大数据教材的问题,我所负责的南京大数据研究院,联合金陵科技学院、河南大学、云创大数据、中国地震局等多家单位,历时两年,编著出版了适合本科教学的《大数据》《大数据库》《大数据实验手册》《数据挖掘》《大数据可视化》《深度学习》等教材。在大数据教学中,本科院校的实践教学应更具系统性,偏向新技术的应用,且对工程实践能力要求更高;而高职高专院校则更偏向于技术性和技能训练,理论以够用为主,学生将主要从事数据清洗和运维方面的工作。基于此,我还联合多家高职院校专家准备了《云计算导论》《大数据导论》《数据挖掘基础》《R语言》《数据清洗》《大数据系统运维》《大数据实践》系列教材。

此外,这些图书的配套PPT和其他资料也继续在中国大数据(thebigdata.cn)和中国

云计算(chinacloud.cn)等网站免费提供。同时,大数据实验平台(<http://bd.cstor.cn>)、免费的物联网大数据托管平台万物云(wanwuyun.com)和环境大数据免费分享平台环境云(envicloud.cn)将持续开放,使资源与数据随手可得,让大数据学习变得更加轻松。

在此,特别感谢我的硕士生导师谢希仁教授和博士生导师李三立院士。谢希仁教授所著的《计算机网络》已经更新到第7版,与时俱进且日臻完善,时时提醒学生要以这样的标准来写书。李三立院士是留苏博士,为我国计算机事业做出了杰出贡献,曾任国家攀登计划项目首席科学家。他严谨治学,带出了一大批杰出的学生。

本丛书是集体智慧的结晶,在此谨向付出辛勤劳动的各位作者致敬!书中难免会有不当之处,请读者不吝赐教。我的邮箱:gloud@126.com,微信公众号:刘鹏看未来(lpoutlook)。

刘 鹏

于南京大数据研究院

# 前 言

---

随着机器学习的兴起和数据科学的应用发展，Python 逐步成了最受欢迎的语言之一。它简单易用、逻辑明确并拥有海量的扩展包，因此不仅成为机器学习与数据科学的首选语言，同时在网页、数据爬取和科学研究等方面也成为不二选择。

《Python 程序设计》是中国信息协会大数据分会副会长刘鹏教授组织编撰的“高级大数据人才培养丛书”之一。一是考虑程序设计要注重实际应用开发，二是由于所在丛书中《数据挖掘》《大数据》《云计算》等对关联规则、神经网络、推荐系统、云计算算法有详细的介绍，故本书没有再介绍有关这几部分的理论及技术。另外，本书的全部实验均可在大数据实验平台 (<http://bd.cstor.cn>) 上远程开展，也可在高校部署的 BDRack 大数据实验一体机上本地开展。

全书分为 12 章，其主要内容如下：

第 1 章“Python 基础”。主要介绍 Python 的特点，Python 的安装与运行、程序控制等。

第 2 章“数据类型”。学习 Python 提供的基本数据类型：整型、浮点型、列表、元组、字典、字符串等。

第 3 章“文件”。学习如何使用 Python 程序对文件进行操作，主要包括如何读写文件、如何处理 Word 文件、如何处理.pdf 文件及压缩文件等。

第 4 章“程序调试”。主要学习程序运行时发生错误或异常的各种处理方法，以及修复程序 bug 的各种调试手段等。

第 5 章“面向对象程序设计”。结合 Python 学习面向对象程序设计，主要包括面向对象程序技术的基本概念、类的定义和对象、类属性、类的方法、类的继承性与多态性等。

第 6 章“连接数据源”。主要学习如何基于 Python 第三方库 pandas 处理 CSV 数据源、Excel 数据源、JSON 数据源，以及数据库的操作。

第 7 章“网络爬虫”。主要学习如何使用 Python 网络爬虫为特定用户准备数据资源，并以热门电影搜索、大数据相关论文文章标题采集、全国空气质量数据爬取为案例进行爬虫项目实战。

第 8 章“数据挖掘”。学习如何用 Python 数据分析工具进行数据挖掘，主要包括数

据预处理、分类与预测、聚类分析，并以信用评估、影片推荐系统等进行数据挖掘项目实战。

第 9 章“自然语言处理”。学习 Python 在自然语言处理方面的应用，主要包括如何应用 NLTK、jieba 完成分词、词性标注、命名实体识别及语法分析等，并以搜索引擎为例进行自然语言处理项目实战。

第 10 章“数据可视化”。学习如何使用 Python 图形库进行绘图操作，实现数据的可视化，主要包括 Pillow、Matplotlib、Echarts 的使用等。

第 11 章“Web 和移动应用”。结合案例学习如何基于 Django 进行 Python Web 开发，以及如何基于 Python Kivy 开发 Python 移动应用。

第 12 章“与云结合”。主要介绍国内各种云服务平台，以及如何运用 Python API、Python SDK 实现访问。

本书框架和内容主要由刘鹏教授规划，第 1~4 章由唐万梅编写，第 5~7、10 章由景雪琴编写，第 8、9、11、12 章由张雪萍编写，全书由张雪萍统稿润色。

本书非常适合作为高校教材使用。自 2012 年始，本人所在学院引进的外方课程——计算机程序设计就是用的 Python 语言，2016 年本人到美国访学期间了解到，美国几乎所有大学非计算机专业都设有 Python 程序设计课程。因此，建议高校为计算机及其相关专业开设 Python 程序设计课程。

本书力求系统全面、通俗易懂，且每章均有习题，可作为高等院校计算机及相关专业的本科和研究生教材。高职高专院校也可以选用本书部分内容开展教学。课程教学建议为 60 学时，其中，上机实验设 16~24 学时为宜。

感谢丛书主编刘鹏教授和金陵科技学院张燕副校长的大力支持和帮助，感谢云创大数据武郑浩老师的辛勤付出，感谢云创大数据沈大伟、保磊老师的技术支持，感谢诸位审稿专家的不吝赐教，感谢诸位编委的鼎力相助。

感谢我的研究生王军峰，第 8、9、11、12 章的实验项目由他完成实际上机验证，这为在大数据实验平台上开展教学实践奠定了基础。

由于编写时间仓促，水平所限，书中难免会出现一些错误或不准确的地方，恳请读者批评指正。如果您有宝贵意见，可通过微信号 [zz67789875](https://www.zz67789875.com) 或邮箱 [zhang\\_xpcn@aliyun.com](mailto:zhang_xpcn@aliyun.com) 联系我。期待在技术之路上与您互勉共进。

张雪萍

于河南工业大学

# 目 录

第 1 章 Python 基础 .....	1
1.1 Python 简介 .....	1
1.2 Python 的安装与运行 .....	3
1.3 Python 版本的选择 .....	8
1.4 程序控制 .....	11
1.4.1 Python 赋值语句 .....	11
1.4.2 顺序结构 .....	18
1.4.3 选择结构 .....	19
1.4.4 循环结构 .....	22
1.5 绘图 .....	27
1.5.1 创建 turtle 对象 .....	27
1.5.2 turtle 绘图的基础知识 .....	29
1.5.3 利用 turtle 库提供的方法绘制图形 .....	31
1.6 函数 .....	37
1.6.1 函数的定义 .....	37
1.6.2 函数的调用 .....	39
1.6.3 lambda 函数 .....	42
习题 .....	42
参考文献 .....	43
第 2 章 数据类型 .....	44
2.1 核心内置数据类型概述 .....	44
2.2 数字类型声明及基本运算 .....	47
2.2.1 整型 .....	47
2.2.2 浮点型 .....	47
2.2.3 复数类型 .....	47
2.2.4 数字运算符 .....	48

2.2.5	数字类型的常用函数及 math 库	49
2.2.6	数字类型转换函数	51
2.2.7	浮点型精度处理	52
2.3	列表	53
2.3.1	列表基本特征	53
2.3.2	序列通用操作	55
2.3.3	可变序列及列表通用操作（一）	58
2.3.4	可变序列及列表通用操作（二）	59
2.4	元组	62
2.5	range	64
2.6	哈希运算	66
2.7	字典	67
2.7.1	字典概述及声明	67
2.7.2	字典元素的访问	71
2.7.3	字典常用方法	71
2.8	字符串	74
2.8.1	字符串的声明	74
2.8.2	转义字符	76
2.8.3	字符串序列通用操作	77
2.8.4	字符串常用内置方法	78
	习题	82
	参考文献	83
<b>第 3 章</b>	<b>文件</b>	<b>84</b>
3.1	读写文件	84
3.1.1	文件对象声明与基本操作	84
3.1.2	编码问题	89
3.1.3	文件写入操作	90
3.1.4	列表推导式	92
3.1.5	关闭文件	94
3.1.6	上下文语法	94
3.1.7	生成器	95
3.2	遍历目录树	97
3.3	处理 Word 文件	102
3.3.1	Python-docx 库	102
3.3.2	利用 Python-docx 库读 Word 文件	102
3.3.3	利用 docx 创建 Word 文件	104

3.4 处理.pdf 文件 .....	105
3.5 处理压缩文件 .....	107
习题 .....	111
参考文献 .....	112
<b>第 4 章 程序调试 .....</b>	<b>113</b>
4.1 异常 .....	113
4.2 断言 .....	127
4.3 日志 .....	131
4.4 调试器 .....	138
习题 .....	144
参考文献 .....	144
<b>第 5 章 面向对象程序设计 .....</b>	<b>146</b>
5.1 面向对象程序技术的基本概念 .....	146
5.2 类的定义和对象 .....	148
5.3 构造函数和析构函数 .....	150
5.4 类属性和实例属性 .....	151
5.5 类的方法 .....	152
5.5.1 类方法 .....	152
5.5.2 实例方法 .....	152
5.5.3 静态方法 .....	152
5.5.4 类的特殊方法 .....	154
5.6 类的继承性 .....	160
5.6.1 单一继承 .....	160
5.6.2 多重继承 .....	162
5.7 类的多态性 .....	164
习题 .....	164
参考文献 .....	165
<b>第 6 章 连接数据源 .....</b>	<b>166</b>
6.1 导入 CSV 数据 .....	166
6.1.1 CSV 数据的格式 .....	166
6.1.2 Python 读取 CSV 文件 .....	167
6.1.3 Python 写 CSV 文件 .....	169
6.2 导入 Excel 数据 .....	170
6.2.1 Python 读取 Excel 文件 .....	171
6.2.2 Python 写 Excel 文件 .....	171

6.3	导入 JSON 数据	172
6.3.1	JSON 数据的格式	172
6.3.2	Python 解码 JSON 数据	172
6.3.3	Python 编码 JSON 数据	173
6.3.4	Python 处理 JSON 数据文件	174
6.4	访问数据库	175
6.4.1	数据库的查询操作	177
6.4.2	数据库的插入操作	178
6.4.3	数据库的删除操作	178
6.4.4	数据库的修改操作	179
	习题	179
	参考文献	180
<b>第 7 章</b>	<b>网络爬虫</b>	<b>181</b>
7.1	网络爬虫工作的基本原理	181
7.1.1	网页的概念	181
7.1.2	网络爬虫的工作流程	182
7.1.3	Python 与网络爬虫	183
7.2	网页内容获取——requests 库	183
7.2.1	requests 对象	184
7.2.2	response 对象	184
7.3	网页内容解析——BeautifulSoup 库	185
7.3.1	BeautifulSoup 库概述	185
7.3.2	BeautifulSoup 库常用方法和 Tag 节点	187
7.4	正则表达式	188
7.4.1	正则表达式概念	188
7.4.2	正则表达式元字符介绍	190
7.4.3	正则表达式的常用函数介绍	194
7.5	实战：热门电影搜索	195
7.6	实战：大数据相关论文文章标题采集	196
7.7	实战：全国空气质量数据爬取	198
	习题	199
	参考文献	200
<b>第 8 章</b>	<b>数据挖掘</b>	<b>201</b>
8.1	Python 常用数据分析工具	201
8.1.1	NumPy	201
8.1.2	Scipy	202

8.1.3	pandas	202
8.1.4	Scikit-Learn	203
8.2	数据预处理	204
8.2.1	数据清理	204
8.2.2	数据集成	205
8.2.3	数据变换	206
8.2.4	Python 数据预处理	206
8.3	分类与预测	210
8.3.1	特征选择	210
8.3.2	性能评估	210
8.3.3	实现过程	212
8.3.4	分类与预测的常用方法	213
8.4	聚类分析	235
8.4.1	聚类分析定义	235
8.4.2	聚类分析评价标准	235
8.4.3	数据相似度度量	236
8.4.4	聚类分析的常用方法	237
8.5	实战: 信用评估	247
8.5.1	数据加载及说明	247
8.5.2	数据预处理	248
8.5.3	划分数据集	249
8.5.4	模型建立及参数调优	250
8.5.5	模型测试及分析	252
8.6	实战: 影片推荐系统	255
8.6.1	推荐系统	255
8.6.2	python-recsys 库简介	256
8.6.3	影片推荐	256
	习题	258
	参考文献	258
<b>第9章 自然语言处理</b>		<b>260</b>
9.1	Python 常用自然语言处理工具	260
9.1.1	Python 自然语言处理工具包 NLTK	260
9.1.2	Python 中文处理工具 jieba	260
9.1.3	Python 语法解析器 PLY	261
9.2	文本处理	261
9.2.1	文本获取	261
9.2.2	文本表示	262

9.2.3	文本特征词提取	263
9.3	词法分析	268
9.3.1	分词	268
9.3.2	词性标注	270
9.3.3	命名实体识别	271
9.3.4	去停用词	272
9.3.5	中文分词实战	275
9.4	语法分析	277
9.4.1	语法分析简介	277
9.4.2	语法树	277
9.4.3	语法分析算法	278
9.4.4	语法分析示例	280
9.5	实战：搜索引擎	282
	习题	290
	参考文献	291
<b>第 10 章</b>	<b>数据可视化</b>	<b>292</b>
10.1	用 Pillow 操作图像	292
10.1.1	图像的基本知识	292
10.1.2	图像处理中常用的模块和函数	293
10.1.3	案例介绍	296
10.2	用 Matplotlib 绘图	298
10.2.1	Matplotlib 常用函数介绍	299
10.2.2	折线图的函数定义及属性说明	299
10.2.3	案例介绍	300
10.3	调用 Echarts	308
	习题	310
	参考文献	310
<b>第 11 章</b>	<b>Web 和移动应用</b>	<b>311</b>
11.1	Web 框架 Django	311
11.1.1	Django 简介	311
11.1.2	Web 框架	311
11.1.3	MVC 和 MTV 模式	312
11.1.4	Django 的安装	313
11.2	Python Web 开发	314
11.2.1	创建项目	314
11.2.2	Django 模板	316

11.2.3 Django 模型 .....	318
11.2.4 Django Admin 管理工具 .....	323
11.2.5 Django Nginx+uwsgi 安装配置 .....	330
11.3 Python 移动应用开发 .....	333
11.3.1 Python Kivy .....	333
11.3.2 Python 移动应用开发过程 .....	334
11.3.3 基于 Python 开发 2048 游戏 .....	335
习题 .....	340
参考文献 .....	341
<b>第 12 章 与云结合 .....</b>	<b>342</b>
12.1 阿里云 .....	342
12.1.1 阿里云计算体系架构 .....	342
12.1.2 CLI Python 版 .....	344
12.2 腾讯云 .....	350
12.2.1 腾讯云总体架构 .....	350
12.2.2 腾讯云 Python 访问 .....	351
12.3 百度云 .....	354
12.3.1 百度云架构 .....	355
12.3.2 BAE Python 部署 .....	357
12.4 万物云 .....	359
12.4.1 功能及应用 .....	360
12.4.2 数据服务及访问 .....	360
12.5 环境云 .....	366
12.5.1 功能服务 .....	366
12.5.2 应用开发数据接口 .....	366
习题 .....	369
参考文献 .....	369
<b>附录 A 人工智能与大数据实验环境 .....</b>	<b>371</b>

# 第 1 章 Python 基础

Python 是“一种解释型的、面向对象的、带有动态语义的高级程序设计语言”<sup>[1,2]</sup>。它能够实现真正的跨平台（可以运行在 Linux、MacOS 和 Windows 上），它的强制缩进的语法使得它的代码简洁易读。对比其他编程语言，Python 更加容易上手。由于 Python 拥有大量第三方库的支持，所以，应用 Python 来进行应用项目的开发更加高效快捷。Python 广泛应用于系统管理工作，Industrial Light & Magic（工业光魔）公司在高预算影片中使用 Python 制作影片的特效，Yahoo! 使用它（包括其他技术）管理讨论组，Google 用它实现网络爬虫和搜索引擎中的很多组件。Python 也被用于计算机游戏和生物信息等各个领域<sup>[2]</sup>。随着人工智能的快速发展，Python 的应用得到了更好的普及。

## 1.1 Python 简介

Python 的创始人是荷兰的吉多·范罗苏姆（Guido van Rossum）。1989 年感恩节期间，吉多为了打发圣诞节的无趣，决心开发一个新的脚本解释程序，作为 ABC 语言的一种继承。之所以选中 Python 作为这门语言的名字，是因为他是 BBC 电视剧《“蒙提·派森”飞行马戏团》（*Monty Python's Flying Circus*）的爱好者。他想营造一种编程语言的神秘感，所以把它命名为 Python。

Python 语言诞生于 1989 年，但第 1 个公开发行版本发行于 1991 年，2000 年 10 月 Python2.0 正式发布，2008 年 12 月 Python3.0 正式发布，目前的最新版本是 Python3.6.5。

Python 是一种面向对象、直译式计算机程序设计语言，也是一种功能强大的通用程序设计语言。它包含了一组完善且容易理解的标准库，并且还有大量第三方库的支持，能够轻松完成很多常见的任务。它的语法非常简捷、清晰，与其他大多数计算机程序设计语言不一样，它采用强制缩进来定义语句块<sup>[1]</sup>。

**Python 的设计风格：**Python 在设计上坚持清晰划一的风格，这使得 Python 成为一门易读、易维护，并且被大量用户所喜欢的、用途广泛的语言。设计者开发时总的指导思想是，对于一个特定的问题，只要有一种最好的方法来解决就好了。

**Python 的设计定位：**Python 的设计哲学是“优雅、明确、简单、可读性强”。

**Python 的面向对象：**Python 是完全面向对象的语言。函数、模块、数字、字符串都是对象，并且完全支持继承、重载、派生、多继承，有益于增强源代码的复用性。Python 支持重载运算符和动态类型。

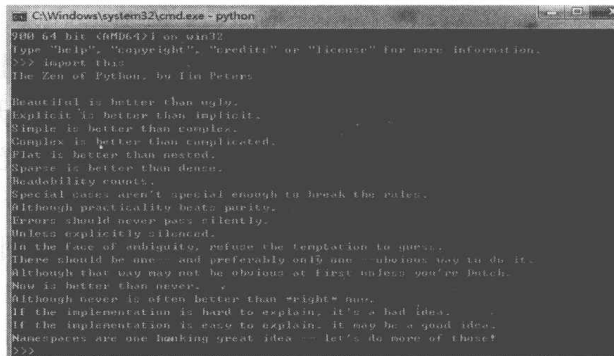
Python 开发者的哲学是“用一种方法，最好是只有一种方法来做一件事”。这一点跟其他大多数的编程语言不太一样，当你使用 Python 语言写程序面临很多种选择时，

Python 的开发者通常会拒绝那些比较花哨的方法，而选择明确的、很少或没有歧义的语法，这些准则就是我们平时所说的 Python 的格言<sup>[3]</sup>。

执行命令“import this”后，你可以看到一篇由 Tim Peters 撰写的文章。它介绍了编写优美的 Python 程序所需要关注的一些重要原则，以此了解 Python 的设计哲学。另外，也可以参考相关网站了解 Python 的设计哲学<sup>[4]</sup>。

```
> import this
```

执行“import this”命令后的显示结果如图 1-1 所示。



```

C:\Windows\system32\cmd.exe - python
Python 2.7.10 Shell
Python 2.7.10 Shell on win32
Type "help()", "copyright()", "credits()" or "license()" for more information.
>>> import this
The Zen of Python, by Tim Peters

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
Unless explicitly silenced.
In the face of ambiguity, refuse the temptation to guess.
There should be one -- and preferably only one -- obvious way to do it.
Although that way may not be obvious at first unless you're Dutch.
Now is better than never.
Although none is often better than "right* now".
If the implementation is hard to explain, it's a bad idea.
If the implementation is easy to explain, it may be a good idea.
Namespaces are one honking great idea -- let's do more of those!
>>>

```

图 1-1 执行“import this”命令后的显示结果

Python 是一种解释型、面向对象、动态数据类型的高级程序设计语言。自从 20 世纪 90 年代初 Python 语言诞生至今，它逐渐被广泛应用于处理系统管理任务和 Web 编程，尤其是随着人工智能领域发展的持续升温，Python 已经成为最受欢迎的程序设计语言之一。Python 语言相较于其他编程语言，有以下主要的优势：

(1) 语法简洁而清晰，代码的可读性高。Python 的语法要求强制缩进，用这种强制缩进来体现语句间的逻辑关系，显著提高了程序的可读性。

(2) 开发效率高。由于它简单明确，所以它也是开发效率比较高的一种编程语言。

(3) 跨平台特性。Python 可以真正做到跨平台，比如我们开发的程序可以运行在 Windows、Linux、MacOS 系统下。这是它的可移植性优势。

(4) 大量丰富的库或扩展。Python 常常被昵称为胶水语言，它能够很轻松地把用其他语言编写的各种模块（尤其是 C/C++）轻松地联结在一起。利用这些大量丰富的第三方库，可以很方便地开发我们自己的应用程序。

(5) 代码量少，一定程度上提高了软件质量。由于使用 Python 语言编写的代码量相比别的语言来说小很多，所以说，它出错的概率也要小很多，这在一定程度上也提高了编写的软件的质量。

Python 的用途非常广泛，它可以用在以下方面：

- (1) 网页开发；
- (2) 可视化 (GUI) 界面开发；
- (3) 网络 (可用于网络方面的编程)；
- (4) 系统编程；
- (5) 数据分析；