



高等院校信息类新专业规划教材  
大数据和人工智能技术丛书



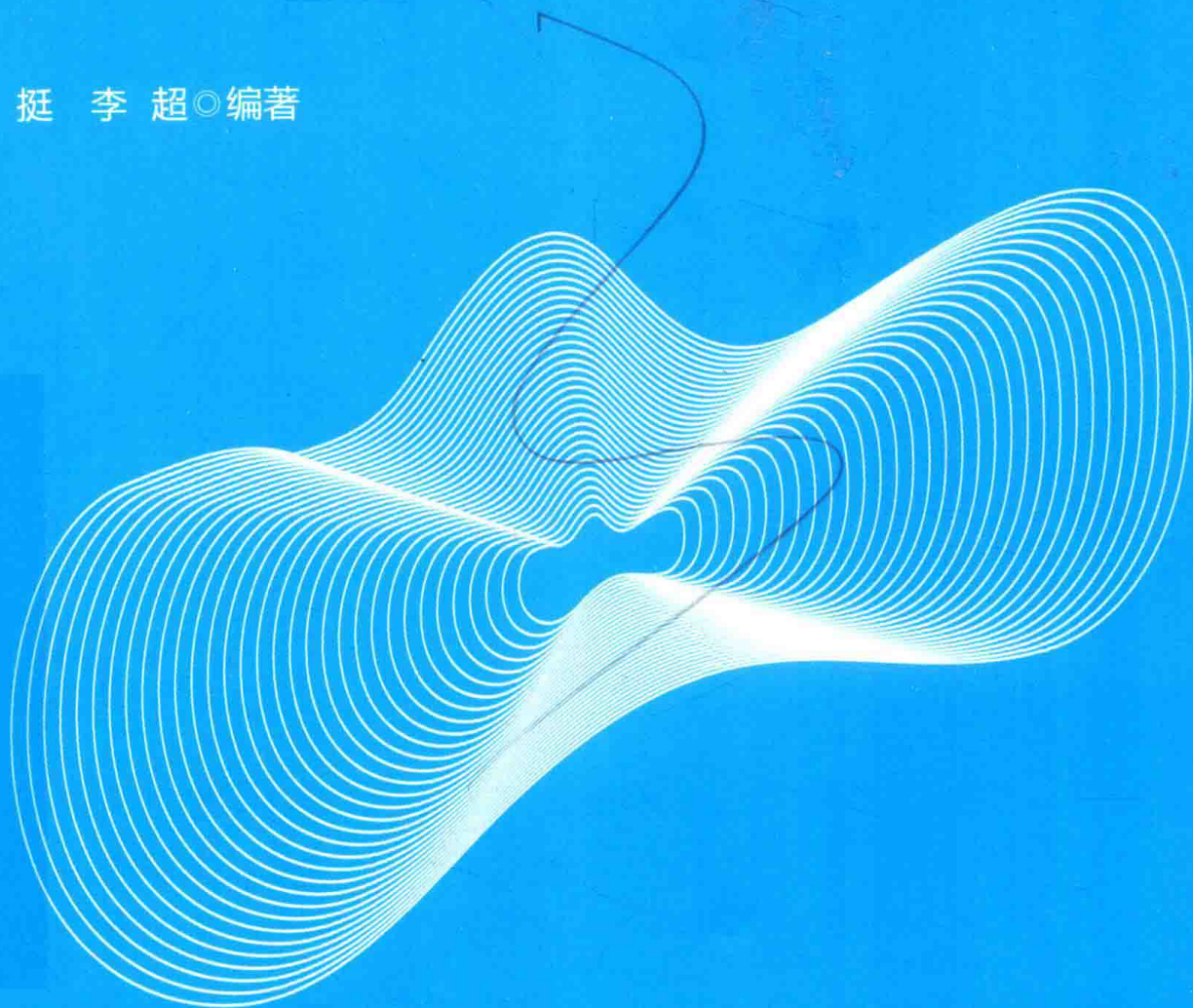
培训推荐教材

华信乾坤

Python DATA ANALYSIS

# 基于Python语言的 数据分析

余挺 李超◎编著



北京邮电大学出版社  
[www.buptpress.com](http://www.buptpress.com)



高等院校信息类新专业规划教材  
大数据和人工智能技术丛书

# 基于 Python 语言的数据分析

余挺 李超 编著



北京邮电大学出版社  
[www. buptpress. com](http://www.buptpress.com)

## 内 容 简 介

本书从 Python 语言语法基础入手,旨在培养读者的基于 Python 语言的数据分析能力。全书共 6 章:第 1 章介绍 Python 的准备工作;第 2 章主要介绍 Python 程序基础;第 3 章介绍 Python 的科学计算基础模块 NumPy;第 4 章讲解 Python 的数据分析基础模块 Pandas;第 5 章介绍 Python 的可视化模块 Matplotlib;第 6 章讲解 Python sklearn 机器学习模块。本书每章都包含演示案例与作业,通过习题和操作实践,读者可以巩固所学的内容。本书适合对 Python 感兴趣的读者阅读,还可作为高等院校 Python 语言相关课程的参考书籍。

### 图书在版编目(CIP)数据

基于 Python 语言的数据分析 / 余挺, 李超编著. -- 北京: 北京邮电大学出版社, 2021. 5

ISBN 978-7-5635-6341-8

I. ①基… II. ①余… ②李… III. ①软件工具—程序设计—高等学校—教材 IV. ①TP311.561

中国版本图书馆 CIP 数据核字(2021)第 035902 号

策划编辑:姚 顺 刘纳新 责任编辑:王晓丹 米文秋 封面设计:七星博纳

出版发行:北京邮电大学出版社

社 址:北京市海淀区西土城路 10 号

邮政编码:100876

发行部:电话:010-62282185 传真:010-62283578

E-mail: publish@bupt.edu.cn

经 销:各地新华书店

印 刷:唐山玺诚印务有限公司

开 本:787 mm×1 092 mm 1/16

印 张:15.75

字 数:412 千字

版 次:2021 年 5 月第 1 版

印 次:2021 年 5 月第 1 次印刷

ISBN 978-7-5635-6341-8

定 价:45.00 元

· 如有印装质量问题,请与北京邮电大学出版社发行部联系 ·

# 前 言

Python 语言自诞生至今经历了约 30 年，但是在前 20 年里，国内使用 Python 进行软件开发的程序员并不多，而在近几年，人们对 Python 语言的关注度迅速提升，这不仅是因为 Python 语言非常优秀，也是因为当下科学计算、人工智能、大数据和区块链等新技术的发展需要。Python 语言具有丰富的动态特性、简单的语法结构和面向对象的编程特点，并且拥有成熟而丰富的第三方库，因此适用于很多领域的软件和硬件开发。

Python 是一门动态的、面向对象的脚本语言，也是一门简约、通俗易懂的编程语言。Python 入门简单，代码可读性强，Python 的这种特性称为“伪代码”，它可以使用户只关心完成什么样的工作任务，而不纠结于 Python 的语法。

另外，Python 是开源的，它拥有非常多优秀的库，可以用于数据分析及其他领域。更重要的是，Python 与最受欢迎的开源大数据平台 Hadoop 和 Spark 具有很好的兼容性。因此，学习 Python 对于有志于向大数据分析岗位发展的数据分析师来说，是一件非常节省学习成本的事。

Web 开发一直是 Python 比较重要的应用场景之一，Python、PHP 和 Java 也是传统 Web 开发的三大解决方案。Python 与 Java 相比具有开发周期短和调整方便的优势，但是 Java 语言在扩展性和性能方面具有一定的优势。PHP 目前在 Web 开发领域占据着较大的开发份额，Python 与 PHP 相比，优点在于代码的灵活性，而 PHP 则有健全的 Web 开发生态，平台兼容性做得非常好。

Python 在大数据和人工智能领域的广泛应用是近些年来 Python 语言取得快速成长的关键因素，Python 广泛应用于大数据应用开发和大数据分析领域，另外，大数据运维也可以使用 Python 语言，相信在大数据落地应用的过程中，Python 语言的使用范围将不断扩大。Python 在人工智能领域的应用涉及自然语言处理、计算机视觉和机器学习等领域，虽然人工智能尚处在行业发展的初期，但是人工智能领域的发展潜力还是非常大的，这也会带动 Python 语言全面拓展自身的应用边界。

Python 在嵌入式领域的应用使得 Python 语言打通了整个物联网开发体系，从设备、网络、平台到分析和应用，整个物联网开发体系都可以采用 Python 语言来完成功能开发，相

信在 5G 通信的推动下，未来 Python 在物联网领域的应用前景也非常值得期待。

本书共分为 6 章，第 1 章介绍学习 Python 的准备工作，包括 Python 的由来与发展、Python 环境搭建、编辑器介绍与安装等；第 2 章主要介绍 Python 的数据类型、程序流程控制语句、函数等内容；第 3 章介绍 Python 的科学计算基础模块 NumPy，包括 NumPy 数组的基本操作、广播机制、索引和读写；第 4 章讲解 Python 的数据分析基础模块 Pandas，包括 Series 和 DataFrame 的使用；第 5 章介绍 Python 的可视化模块 Matplotlib，包括常用图形和 3D 图形的制作；第 6 章讲解 Python sklearn 机器学习模块，包括监督学习、无监督学习，从算法理论分析到案例演示，讲解回归、分类、聚类、降维等常用的机器学习算法。本书每章都包含演示案例与作业，通过习题和操作实践，读者可以巩固所学的内容。通过阅读本书，读者将迅速掌握编程概念，打下坚实的基础，并养成良好的习惯，进而可以开始学习 Python 高级技术，同时能够更轻松地掌握其他编程语言。

继续使用 Python，还是转而使用其他语言？相信读者都会考虑这个问题。可笔者依然推荐 Python，其中的原因有很多。Python 是一门效率极高的语言：相比于其他语言，使用 Python 编写时，程序包含的代码行更少。Python 的语法有助于创建整洁的代码：相比于其他语言，使用 Python 编写的代码更容易阅读、调试和扩展。

Python 可应用于众多方面：编写游戏、创建 Web 应用程序、解决商业问题以及供各类公司开发内部工具，Python 还在科学领域被大量用于学术研究和应用研究。

编程人员依然使用 Python 的一个最重要的原因是，Python 社区有形形色色充满激情的人。对程序员来说，社区非常重要，因为编程绝非孤独的修行，大多数程序员都需要向解决过类似问题的人寻求建议，经验最为丰富的程序员也不例外。需要有人帮助解决问题时，有一个联系紧密、互帮互助的社区至关重要，而对将 Python 作为第一门计算机编程语言来学习的人而言，Python 社区无疑是坚强的后盾。

Python 是一门杰出的语言，值得读者去学习，咱们现在就开始吧！

# 目 录

<b>第 1 章 Python 开发环境</b> .....	1
1.1 Python 语言概述 .....	1
1.1.1 Python 的发展历程 .....	1
1.1.2 Python 的应用场景 .....	2
1.2 搭建编程环境 .....	3
1.2.1 Python 2 和 Python 3 .....	4
1.2.2 Hello World 程序 .....	4
1.3 在不同操作系统中搭建 Python 编程环境 .....	4
1.3.1 在 Linux 系统中搭建 Python 编程环境 .....	4
1.3.2 在 OS X 系统中搭建 Python 编程环境 .....	5
1.3.3 在 Windows 系统中搭建 Python 编程环境 .....	5
1.4 从终端运行 Python 程序 .....	6
1.4.1 在 Linux 和 OS X 系统中从终端运行 Python 程序 .....	6
1.4.2 在 Windows 系统中从终端运行 Python 程序 .....	7
1.5 使用开发软件部署 Python 环境 .....	7
1.5.1 Anaconda 的下载和安装 .....	7
1.5.2 PyCharm 的下载安装及配置 .....	11
1.6 本章小结 .....	13
1.7 本章作业 .....	13
<b>第 2 章 Python 程序基础</b> .....	14
2.1 变量 .....	14
2.1.1 变量的命名和使用 .....	15
2.1.2 使用变量时避免命名错误 .....	15
2.2 字符串 .....	15

2.2.1	修改字符串的大小写	16
2.2.2	合并(拼接)字符串	16
2.2.3	Python 2 中的 print 语句	16
2.3	数字	17
2.3.1	整数	17
2.3.2	浮点数	18
2.3.3	复数	18
2.4	列表	19
2.4.1	访问列表元素	19
2.4.2	列表索引	19
2.4.3	修改、添加和删除元素	20
2.4.4	列表排序	22
2.4.5	列表切片	23
2.4.6	列表复制	26
2.5	元组	29
2.5.1	定义元组	29
2.5.2	修改元组变量	29
2.6	字典	30
2.6.1	字典的创建	30
2.6.2	字典元素的操作	32
2.6.3	字典的常用方法	33
2.6.4	使用字典格式化字符串	38
2.7	本章小结	39
2.8	本章作业	39
<b>第 3 章</b>	<b>Python NumPy</b>	<b>41</b>
3.1	NumPy 简介	41
3.1.1	ndarray 对象	41
3.1.2	数据类型	43
3.1.3	数组属性	45
3.1.4	数组创建	47
3.1.5	切片和索引	54
3.1.6	高级索引	57

3.1.7	矩阵变形	61
3.1.8	添加/删除元素	71
3.1.9	字符串函数	77
3.1.10	算术函数	81
3.1.11	统计函数	83
3.1.12	排序、搜索和计数函数	89
3.1.13	线性代数	94
3.2	本章小结	99
3.3	本章作业	99
<b>第 4 章</b>	<b>Python Pandas</b>	<b>100</b>
4.1	Pandas 基础	100
4.1.1	Pandas 模块安装	100
4.1.2	Pandas 数据结构	101
4.1.3	系列	102
4.1.4	数据帧	107
4.1.5	面板	115
4.1.6	Pandas 常见功能	118
4.1.7	Pandas 描述性统计	123
4.1.8	Pandas 排序	128
4.2	Pandas 数据分析	131
4.2.1	Pandas 统计函数	131
4.2.2	Pandas 聚合函数	134
4.2.3	Pandas 分组功能	140
4.2.4	Pandas 数据连接查询	146
4.3	Pandas 数据清洗	151
4.3.1	缺失值处理	151
4.3.2	异常值处理	156
4.3.3	重复值处理	159
4.4	本章小结	160
4.5	本章作业	161
<b>第 5 章</b>	<b>Matplotlib 绘图库</b>	<b>162</b>
5.1	Matplotlib 的基本使用	162

5.1.1	Matplotlib 简介 .....	162
5.1.2	Matplotlib 的基本属性和方法 .....	163
5.2	Matplotlib 多种图形的绘制 .....	171
5.2.1	散点图绘制 .....	171
5.2.2	饼图绘制 .....	178
5.2.3	条形图 .....	188
5.2.4	直方图 .....	190
5.2.5	其他图形 .....	191
5.2.6	多图合并显示 .....	194
5.3	本章小节 .....	201
5.4	本章作业 .....	202
<b>第 6 章</b>	<b>Python sklearn .....</b>	<b>204</b>
6.1	Python sklearn 模块功能和特点 .....	205
6.2	回归算法 .....	206
6.2.1	线性回归 .....	206
6.2.2	逻辑回归 .....	213
6.3	分类算法 .....	213
6.3.1	K 近邻算法 .....	214
6.3.2	决策树算法 .....	217
6.3.3	朴素贝叶斯算法 .....	220
6.3.4	支持向量机算法 .....	223
6.4	聚类算法 .....	224
6.5	降维算法 .....	231
6.5.1	主成分分析 .....	232
6.5.2	线性判别分析 .....	237
6.6	本章小结 .....	242
6.7	本章作业 .....	242
<b>参考文献</b>	<b>.....</b>	<b>243</b>

# 第 1 章

## Python 开发环境

---

### 1.1 Python 语言概述

Python 是一种跨平台的计算机程序设计语言,也是一种面向对象的动态类型语言,最初被设计用于编写自动化脚本(shell),随着版本的不断更新和语言新功能的添加,其越来越多地被用于独立的大型项目的开发。

Python 是目前最流行的全场景编程语言之一,在 Web 开发、大数据开发、人工智能开发和嵌入式开发等领域均有广泛的应用,同时 Python 语言是网络运维相关人员比较常用的编程工具之一。由于 Python 语言自身的语法结构比较简单易学,而且有丰富的开发库支撑,因此未来 Python 的应用前景是非常广阔的。

#### 1.1.1 Python 的发展历程

自 20 世纪 90 年代初 Python 语言诞生至今,它已被广泛应用于系统管理任务的处理和 Web 编程。Python 的创始人为荷兰人 Guido van Rossum。1989 年圣诞节期间,在阿姆斯特丹,Guido 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。Guido 选中 Python(大蟒蛇的意思)作为该编程语言的名字,其取自英国 20 世纪 70 年代首播的电视喜剧《蒙提·派森的飞行马戏团》(*Monty Python's Flying Circus*)。ABC 是由 Guido 参加设计的一种教学语言,就 Guido 本人看来,ABC 这种语言非常优美和强大,是专门为非专业程序员设计的,但是 ABC 语言并没有成功,究其原因,Guido 认为是 ABC 的非开放性造成的。Guido 决心在 Python 中避免这一错误,同时,他还想实现在 ABC 中闪过过但未曾实现的东西。

就这样,Python 在 Guido 手中诞生了。可以说,Python 是从 ABC 发展起来的,主要受到了 Modula-3(另一种相当优美且强大的语言,是为小型团体设计的)的影响,并且结合了 Unix shell 和 C 的习惯。

Python 目前已经成为最受欢迎的程序设计语言之一。自 2004 年开始,Python 的使用率呈线性增长,Python 2 于 2000 年 10 月 16 日发布,目前稳定版本是 Python 2.7,Python 3 于 2008 年 12 月 3 日发布,不完全兼容 Python 2。2011 年 1 月,Python 被 TIOBE 编程语言排行榜评为 2010 年度语言。

## 1.1.2 Python 的应用场景

如果你想学习 Python,或者你刚开始学习 Python,那么你可能会问:“我能用 Python 做什么?”这个问题其实并不好回答,因为 Python 有很多用途。

大家都知道,当下互联网全栈软件开发工程师的概念很火,而 Python 可以作为一种全栈开发语言,所以如果你能学好 Python 语言,那么前端开发、后端开发、软件测试、大数据分析 & 挖掘、网络爬虫等工作你都能胜任。综合企业的应用场景,目前 Python 开发主要有以下四大应用:

- 网络爬虫;
- Web 开发;
- 人工智能;
- 自动化运维。

### 1. 网络爬虫

网络爬虫又称网络蜘蛛,是指按照某种规则在网络上爬取网页所需内容的脚本程序。众所周知,每个网页通常都包含其他网页的入口,网络爬虫则通过一个网址依次进入其他网址获取所需内容。爬虫的作用如下所述。

- 做垂直搜索引擎:Google 和 Baidu 使用 Python 语言作为通用搜索引擎网页收集器。
- 科学研究:在线人类行为、在线社群演化、人类动力学研究、计量社会学、复杂网络、数据挖掘等领域的实证研究都需要大量数据。
- 网络爬虫:Python 语言是收集相关数据的利器。

爬虫是搜索引擎的第一步,也是最容易的一步。用什么语言写爬虫?大家可能熟悉 C、C++,其优点是效率高,快速,适合通用搜索引擎做全网爬取,其缺点是开发慢,代码非常长,如天网搜索引擎的源代码。

脚本语言:Perl、Python、Java、Ruby 等。相对简单、易学,良好的文本处理能方便网页内容的细致提取,但效率往往不高,适合对少量网站的聚焦爬取。

为什么最终选择 Python?其实多种爬虫语言区别不大,原理就是利用好正则表达式。很多爬虫程序都是用 Python 写的,通俗易懂且简单明了。Python 的优势很多,以下为两个要点。

#### (1) 抓取网页本身的接口

相对于其他静态编程语言,如 Java、C#、C++,Python 抓取网页文档的接口更简洁;相对于其他动态脚本语言,如 Perl、Shell,Python 的 urllib2 包提供了较为完整的访问网页文档的应用程序接口(API)。

此外,抓取网页有时候需要模拟浏览器的行为,很多网站会封杀生硬的爬虫抓取。这时我们需要模拟 user agent 的行为构造合适的请求,如模拟用户登录、模拟 session/cookie 的存储和设置,这些在 Python 语言中有非常优秀的第三方包(如 Requests 和 lxml 等)帮用户完成。

#### (2) 网页抓取后的处理

抓取的网页通常需要处理,如过滤 HTML 标签、提取文本等。Python 语言的 beautifulsoup 提供了简洁的文档处理功能,能用极短的代码完成大部分文档的处理。其实以上功能很多语言和工具都能实现,但是用 Python 语言能够实现得最快、最干净。

## 2. Web 开发

Web 是一种基于超文本和 HTTP 的、全球性的、动态交互的、跨平台的分布式图形信息系统,是建立在 Internet 上的一种网络服务,为浏览者在 Internet 上查找和浏览信息提供了图形化的、易于访问的直观界面,其中的文档及超级链接将 Internet 上的信息节点组织成一个互相关联的网状结构。

Django 和 Flask 等基于 Python 的 Web 框架在 Web 开发中非常流行,那么使用 Python 开发网站需要用到哪些知识呢?

- ① Python 基础,包括条件判断、循环、函数、类和对象等面向对象开发知识。
- ② HTML 和 CSS 的基础知识以及前端开发相关框架。
- ③ 数据库基础知识,数据库可以作为网站的后台数据存储,加上中间件组合成为综合性的网站服务。

读者掌握上述知识后即可尝试开发一个简单的小站,如果想开发比较大型的、业务逻辑比较复杂的网站,则需要掌握其他的知识,如 Redis 缓存、MQ 中间件等。

## 3. 人工智能

人工智能(AI,Artificial Intelligence)是研究、开发用于模拟、延伸和扩展人的智能的理论、方法、技术及应用系统的一门新的技术科学。人工智能是计算机科学的一个分支,它企图了解智能的实质,并生产出一种新的能以与人类智能相似的方式做出反应的智能机器,该领域的研究包括机器人、语言识别、图像识别、自然语言处理和专家系统等。人工智能自诞生以来,理论和技术日益成熟,应用领域也不断扩大,可以设想,未来人工智能带来的科技产品将会是人类智慧的“容器”,也可能超过人的智能。

对希望加入 AI 和大数据行业的开发人员来说,把鸡蛋放在 Python 这个篮子里不但是安全的,而且是必须的。或者换个方式说,如果将来想在这个行业有所发展,什么都不用想,先把 Python 学会。当然,Python 也存在问题和短处,开发人员可以也应该掌握另外一种甚至几种语言,与 Python 形成搭配,但是 Python 将坐稳数据分析和 AI 第一语言的位置,这一点毫无疑问。笔者认为,由于 Python 将坐稳这个位置,由于这个行业未来需要大批的从业者,更由于 Python 正在迅速成为全球大中小学编程入门课程的首选教学语言,这种开源动态脚本语言非常有机会在不久的将来成为第一种真正意义上的编程世界语。

## 4. 自动化运维

随着技术的进步以及业务需求的快速增长,一个 IT 运维人员通常要管理成千上万台服务器,IT 运维工作也变得重复繁杂。IT 运维工作自动化能够把 IT 运维人员从服务器的管理中解放出来,让运维工作变得简单、快速、准确。Python 运维自动化中的常用模块包括 pexpect、paramiko、nmap、fabric,如今企业中使用最多的两大 DevOps“神器”——Ansible 和 Saltstack,以及运维开发首选的 Python Web 框架 Django,都是将 Python 语言作为基础底层开发语言的。

# 1.2 搭建编程环境

在不同的操作系统中,Python 存在细微的差别,因此有几点读者需要牢记在心。本节将介绍大家使用的两个主要的 Python 版本,并简要介绍 Python 的安装步骤。

## 1.2.1 Python 2 和 Python 3

当前有两个不同的 Python 版本:Python 2 和较新的 Python 3。每种编程语言都会随着新概念和新技术的推出而不断发展,Python 的开发者也一直致力于丰富和强化 Python 的功能。大多数修改是逐步进行的,用户几乎意识不到,但如果用户的系统安装的是 Python 3,那么有些使用 Python 2 编写的代码可能无法正确地运行。在本书中,笔者将指出 Python 2 和 Python 3 的差别,这样用户无论安装的是哪个版本,都能够按本书中的说明去做。

如果系统中安装了两个版本,请使用 Python 3;如果没有安装 Python,请安装 Python 3;如果只安装了 Python 2,也可直接使用 Python 2 来编写代码,但还是尽快升级到 Python 3 为好。

## 1.2.2 Hello World 程序

长期以来,编程界都认为刚接触一门新语言时,如果首先使用它来编写一个在屏幕上显示消息“Hello World!”的程序,将给用户带来好运。要使用 Python 来编写 Hello World 程序,只需一行代码:

```
>>> print("Hello World!")
```

这种程序虽然简单,却有其用途:如果它能够在用户的系统上正确地运行,那么用户编写的任何 Python 程序都将如此。稍后将介绍如何在特定的系统中编写这样的程序。

## 1.3 在不同操作系统中搭建 Python 编程环境

Python 是一种跨平台的编程语言,这意味着它能够在所有主要的操作系统中运行。在所有安装了 Python 的现代计算机上,都能够运行任何 Python 程序。然而,在不同的操作系统中,安装 Python 的方法存在细微的差别。

下面将详细介绍如何在各种操作系统中完成这些任务,让读者能够搭建一个对初学者友好的 Python 编程环境。

### 1.3.1 在 Linux 系统中搭建 Python 编程环境

Linux 系统是为编程而设计的,因此在大多数 Linux 计算机中,都默认安装了 Python。编写和维护 Linux 的人认为,用户很可能会使用这种系统进行编程,他们也鼓励用户这样做。鉴于此,要在 Linux 系统中编程,用户几乎不用安装其他软件,也几乎不用修改设置。

在系统中运行应用程序 Terminal,打开一个终端窗口。为确定是否安装了 Python,执行命令 `python`(请注意,其中的“p”是小写的)。输出结果如下所示,它指出了安装的 Python 版本,最后的“>>>”是一个提示符,让用户能够输入 Python 命令。

```
$ python
Python 2.7.6 (default, Mar 22 2014, 22:59:38)
[GCC 4.8.2] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

上述输出结果表明,当前计算机默认使用的 Python 版本为 Python 2.7.6。看到上述输出结果后,如果要退出 Python 并返回终端窗口,可按“Ctrl+D”键或执行命令 `exit()`。

要检查系统是否安装了 Python 3,可能需要指定相应的版本。换句话说,如果输出结果指出默认版本为 Python 2.7,请尝试执行命令 `python3`:

```
$ python3
Python 3.5.0 (default, Sep 17 2015, 13:05:18)
[GCC 4.8.4] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

上述输出结果表明,系统中也安装了 Python 3,因此用户可以使用这两个版本中的任何一个。在使用 Python 3 的情况下,请将本书中的命令“python”都替换为“python3”。

### 1.3.2 在 OS X 系统中搭建 Python 编程环境

大多数 OS X 系统都默认安装了 Python。确定安装了 Python 后,用户还需安装一个文本编辑器,并确保其配置正确无误。

在文件夹 Applications/Utilities 中,选择 Terminal,打开一个终端窗口,用户也可以按“Command+空格”键,再输入“terminal”并按 Enter 键。为确定是否安装了 Python,请执行命令 `python`(注意,其中的“p”是小写的)。输出结果如下所示,它指出了安装的 Python 版本,最后的“>>>”是一个提示符,让用户能够输入 Python 命令。

```
$ python
Python 2.7.5 (default, Mar 9 2014, 22:15:05)
[GCC 4.2.1 Compatible Apple LLVM 5.0 (clang-500.0.68)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

上述输出结果表明,当前计算机默认使用的 Python 版本为 Python 2.7.5。看到上述输出结果后,如果要退出 Python 并返回终端窗口,可按“Ctrl+D”键或执行命令 `exit()`。

### 1.3.3 在 Windows 系统中搭建 Python 编程环境

Windows 系统并非都默认安装了 Python,因此用户可能需要自行下载并安装 Python,再下载并安装一个文本编辑器。

#### 1. 安装 Python

首先,用户需要检查系统中是否安装了 Python。为此,用户可以在“开始”菜单中输入“command”并按 Enter 键以打开一个命令窗口,用户也可按住 Shift 键并右击桌面,再选择“在此处打开命令窗口”。在命令窗口中输入“python”并按 Enter 键,如果出现了 Python 提示符“>>>”,就说明系统中安装了 Python。

然而用户也可能会看到一条错误消息,指出 `python` 是无法识别的命令。此时就需要下载 Windows Python 安装程序,下载地址为 <http://python.org/downloads/>。

下载页面中有两个按钮,分别用于下载 Python 3 和 Python 2。单击用于下载 Python 3

的按钮,会根据用户的系统自动下载正确的安装程序。下载完成后,运行该安装程序,请务必选中复选框“Add Python to PATH”,如图 1-1 所示,这能使用户更轻松地配置系统。

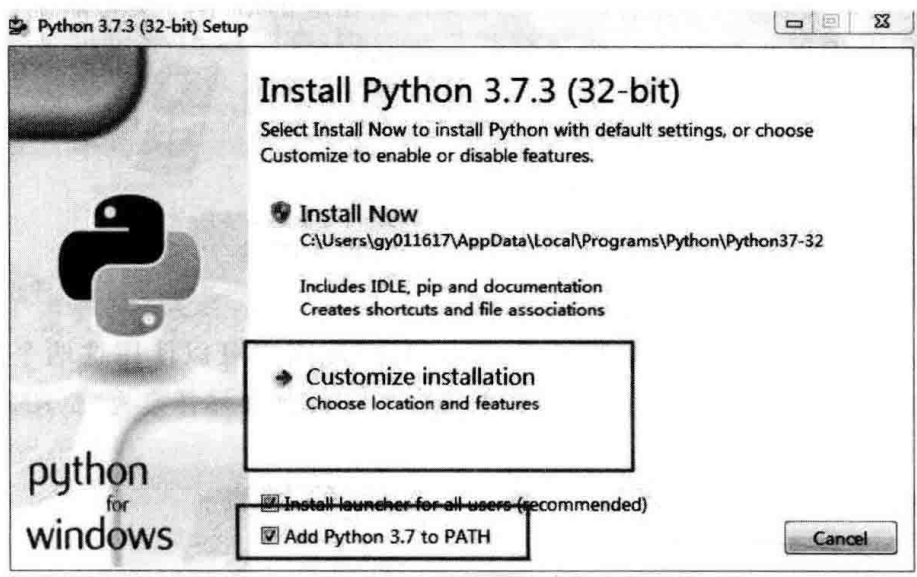


图 1-1 选中复选框“Add Python to PATH”

## 2. 启动 Python 终端会话

通过配置系统,让其能够在终端会话中运行 Python,可简化文本编辑器的配置工作。打开一个命令窗口,并在其中执行命令 python。如果出现了 Python 提示符“>>>”,就说明 Windows 找到了刚安装的 Python 版本。

```
C:\> python
Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015, 22:15:05) [MSC v.1900 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

## 1.4 从终端运行 Python 程序

用户编写的大多数程序都将直接在文本编辑器中运行,但有时候从终端运行程序很有用。例如,用户可以直接运行既有的程序。

在任何安装了 Python 的系统上都可以这样做,前提是要知道如何进入程序文件所在的目录。为尝试这样做,请读者确保已将文件 hello\_world.py 存储在桌面上的 python\_work 文件夹中(保存 Python 代码的文件夹,本书中可以将其作为项目的根文件夹)。

### 1.4.1 在 Linux 和 OS X 系统中从终端运行 Python 程序

在 Linux 和 OS X 系统中,从终端运行 Python 程序的方式相同。在终端会话中,可使用终端命令 cd(change directory,切换目录)在文件系统中导航,可使用命令 ls(list 的简写)显示当前目录中所有未隐藏的文件。

为运行程序 hello\_world.py,请打开一个新的终端窗口,并执行下面的命令:

```

~$ cd Desktop/python_work/
~/Desktop/python_work$ ls
hello_world.py
~/Desktop/python_work$ python hello_world.py
Hello Python World!

```

## 1.4.2 在 Windows 系统中从终端运行 Python 程序

在命令窗口中,可使用终端命令 `cd` 在文件系统中导航,可使用命令 `dir`(directory,目录)列出当前目录中的所有文件。

为运行程序 `hello_world.py`,请打开一个新的终端窗口,并执行下面的命令:

```

C:\> cd Desktop\python_work
C:\Desktop\python_work> dir
hello_world.py
C:\Desktop\python_work> python hello_world.py
Hello Python world!

```

## 1.5 使用开发软件部署 Python 环境

子曰:“工欲善其事,必先利其器。”要学习 Python 就需要有编译 Python 程序的软件,一般情况下,我们选择在 Python 官网下载对应版本的 Python,然后用记事本编写,再在终端进行编译运行即可。但是对于项目级的开发,我们可以选择更好的 Python 开发环境——Anaconda 软件。

Anaconda 是一个基于 Python 的数据处理和科学计算平台,它内置了许多非常有用的第三方库。安装 Anaconda 就相当于把 Python 和一些常用的库(如 NumPy、Pandas、SciPy、Matplotlib 等)自动安装好了,比常规的 Python 安装要容易。如果选择安装 Python,那么还需要 `pip install` 一个一个地安装各种库,安装起来比较烦琐,还需要考虑兼容性。如果非要如此,就要去 Python 官网(<https://www.python.org/downloads/windows/>)选择对应的版本下载安装,可以选择默认安装或者自定义安装,为了避免配置环境和安装 `pip` 的麻烦,如前文所述,建议用户勾选添加 PATH 环境变量和安装 `pip` 选项按钮,使得以后可以不用路径来执行 Python 命令。

### 1.5.1 Anaconda 的下载和安装

**步骤一:**进入官网(<https://www.anaconda.com/download/#windows>)下载对应正确的版本,这里选择的是 Windows 64 bit,如图 1-2 所示。

**步骤二:**双击下载的 exe 文件进行安装,然后单击“Next”,如图 1-3 所示。

**步骤三:**单击“I Agree”,接受安装许可协议,如图 1-4 所示。

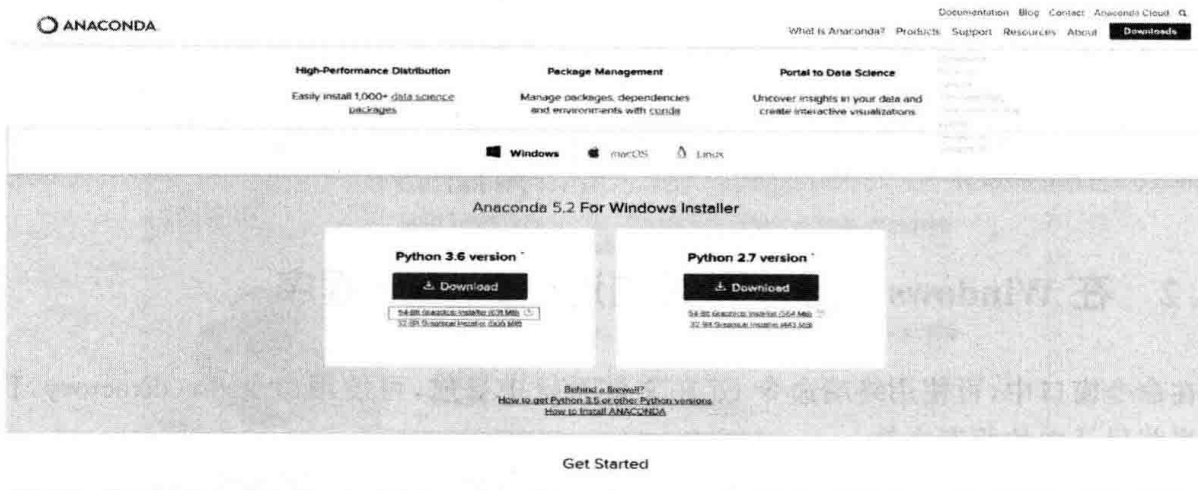


图 1-2 选择 Anaconda 的版本

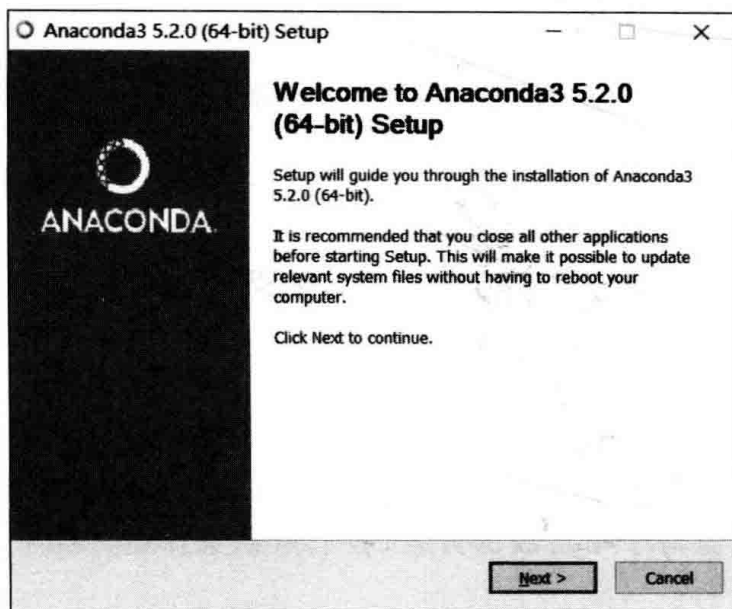


图 1-3 开始安装

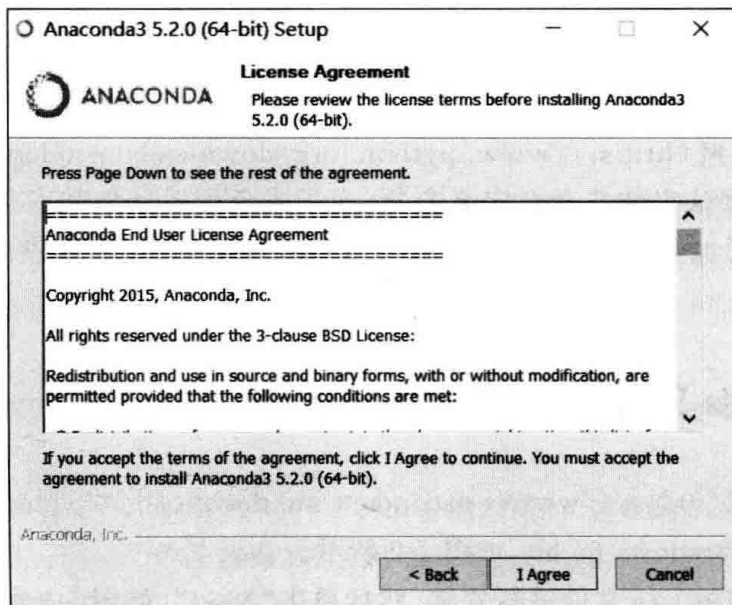


图 1-4 接受安装许可协议