

高等院校计算机基础教育应用型系列规划教材



Python语言 程序设计基础

Python YUYAN CHENGXU SHEJI JICHU

刘晓勇 付 辉 主编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

高等院校计算机基础教育应用型系列规划教材

Python 语言程序设计基础

刘晓勇 付 辉 主编



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

随着数据时代的来临, Python 语言已逐渐成为国内外广泛使用的计算机编程语言之一, 学会使用 Python 语言进行程序设计是从事计算机类工作者的一项基本技能。本书共分 10 章, 比较全面地介绍了 Python 语言的基本语法及编程技巧, 主要内容包括 Python 概述, 数据运算, 程序控制结构, 列表、元组和字典, 函数, 模块, 文件操作, 异常处理, 面向对象程序设计, 常用扩展库等。通过本书的学习, 读者基本上可以掌握 Python 语言的主要内容和使用方法, 同时也可学到程序设计的方法及初级算法的知识, 初步掌握利用计算机编程解决问题的全过程。

本书适合作为高等院校各专业学习 Python 语言的入门教材, 也可作为 Python 语言程序开发人员的自学教材或参考用书。

图书在版编目 (CIP) 数据

Python 语言程序设计基础 / 刘晓勇, 付辉主编. —
北京: 中国铁道出版社, 2019. 1
高等院校计算机基础教育应用型系列规划教材
ISBN 978-7-113-25166-6

I. ①P… II. ①刘… ②付… 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 (2018) 第 271804 号

书 名: Python 语言程序设计基础
作 者: 刘晓勇 付 辉 主编

策 划: 周海燕 韩从付
责任编辑: 周海燕 彭立辉
封面设计: 刘 颖
责任校对: 张玉华
责任印制: 郭向伟

读者热线: (010) 63550836

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

网 址: <http://www.tdpress.com/51eds/>

印 刷: 三河市兴博印务有限公司

版 次: 2019 年 1 月第 1 版 2019 年 1 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 10 字数: 199 千

书 号: ISBN 978-7-113-25166-6

定 价: 32.00 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, 请与本社教材图书营销部联系调换。电话: (010) 63550836

打击盗版举报电话: (010) 51873659

前言

PREFACE

Life is short, you need Python! 这是关于 Python 的一句经典的、富有情怀的话,很多 Python 开发者都是从这句话开始学习、掌握并爱上这门语言的。Python 语法简单,入门非常容易,如果是从零开始学习编程,Python 是一种不错的选择。Python 也被称为是一种“胶水语言”,可以方便地调用其他语言编写的功能模块,并将它们有机地结合在一起形成更高效的新程序。Python 简洁的语法和对动态输入的支持,再加上解释性语言的特性,使得它在许多领域都是一种理想的脚本语言,特别适用于快速应用程序开发。当前,Python 已被广泛应用于众多领域,如:科学计算、数据分析、Web 开发、系统运维、机器学习、人工智能等。正是其开源、简单、易用的特点,近年来吸引了越来越多的开发者使用这门语言。

本书在编写过程中,充分考虑到读者的认知规律,采用通俗易懂的语言,同时考虑培养读者的计算思维能力,辅以较多的案例,启发读者的思维。

全书共分 10 章,其中,第 1 章介绍了 Python 的发展历史及其广泛的应用领域,分析了 Python 自身的优点和不足之处,最后以当前较新的版本为例,介绍了 Python 的安装以及相关开发工具的安装和使用。第 2 章主要介绍了 Python 中的变量、常量及命名规则,几种基本的数据类型,以及 7 种不同操作符和相应的表达式运算。第 3 章介绍了 Python 语言的 3 种常见的程序控制结构,即顺序结构、选择结构和循环结构。第 4 章介绍了 Python 中的列表、元组、字典和集合等几种特殊数据类型,以及相应的基本操作。第 5 章介绍了函数的基本概念、Python 语言中函数的定义和调用过程,以及常用的内置函数。第 6 章介绍了模块的概念以及导入、发布和安装过程。第 7 章介绍了文件的相关操作,包括文件的定义、打开、关闭以及读/写等操作。第 8 章介绍了异常的类型及相关处理方法,包括异常的捕获和处理方法、Python 提供的异常类型,以及通过不同的异常类型来改进程序设计的方法。第 9 章介绍了 Python 面向对象程序设计相关知识,包括类的声明方式以及封装、继承、多态等方面的知识。第 10 章介绍了常用扩展库,如 NumPy、Scipy、Pandas、Matplotlib、Scikitlearn 等。

本书适合作为高等院校各专业学习 Python 语言的入门教材,也可作为 Python 语言程序开发人员的自学教材或参考用书。

本书由刘晓勇、付辉主编,其中第 1 章、第 6~10 章、附录由刘晓勇编写,第 2~5 章由付辉编写。本书在编写过程中,得到广东技术师范学院教务处及计算机科学学院相关领导的大

力支持和鼓励；在出版过程中，中国铁道出版社的编辑付出了艰辛的努力，并给予了无私的帮助，在此一并表示感谢。

由于时间仓促，加之编者能力和学识有限，在编写过程中虽然已经尽了最大努力，但仍难免存在疏漏与不妥之处，恳请读者批评指正。

编者

2018年10月于广州

第 1 章 Python 概述	1	2.3.4 逻辑操作符和表达式	33
1.1 Python 简史	1	2.3.5 位操作符和表达式	33
1.2 Python 的应用领域	2	2.3.6 身份操作符和表达式	34
1.3 Python 的特点	3	2.3.7 成员操作符和表达式	35
1.4 Python 安装及开发环境构建	4	2.3.8 操作符的优先级	36
1.4.1 Python 3.6.5 下载与安装	4	小结	37
1.4.2 Anaconda 下载与安装	8	习题	37
1.4.3 开发工具 Spyder	12	第 3 章 程序控制结构	38
1.4.4 开发工具 PyCharm	13	3.1 程序设计过程	38
小结	19	3.1.1 程序设计结构	38
习题	19	3.1.2 程序流程图	38
第 2 章 数据运算	20	3.2 顺序结构	38
2.1 变量与常量	20	3.2.1 行与缩进	39
2.1.1 标识符	20	3.2.2 语句换行	39
2.1.2 关键字	20	3.2.3 注释	40
2.1.3 转义字符	21	3.2.4 输入语句	40
2.1.4 常量	22	3.2.5 输出语句	41
2.1.5 变量	23	3.2.6 顺序结构举例	43
2.2 基本数据类型	24	3.3 选择结构	44
2.2.1 数值型	24	3.3.1 单分支选择结构	44
2.2.2 布尔型	25	3.3.2 多分支选择结构	45
2.2.3 字符串	25	3.3.3 三元表达式	47
2.2.4 复数类型	28	3.3.4 选择结构举例	47
2.2.5 空值	29	3.4 循环结构	48
2.3 操作符和表达式	30	3.4.1 while 语句	48
2.3.1 算术操作符和表达式	30	3.4.2 for 语句	49
2.3.2 关系操作符和表达式	31	3.4.3 循环嵌套	49
2.3.3 赋值操作符和表达式	32	3.4.4 循环控制语句	50

3.4.5 循环结构举例.....	52	第 5 章 函数.....	74
小结.....	53	5.1 函数的概念.....	74
习题.....	53	5.2 函数的定义.....	74
第 4 章 列表、元组和字典.....	55	5.3 函数的调用.....	76
4.1 列表.....	55	5.3.1 函数调用的形式.....	76
4.1.1 访问列表元素.....	55	5.3.2 函数调用时的数据传递.....	76
4.1.2 更新列表元素.....	56	5.3.3 函数调用的过程.....	77
4.1.3 删除列表元素.....	56	5.4 匿名函数.....	77
4.1.4 列表脚本操作符.....	56	5.5 局部变量和全局变量.....	78
4.1.5 截取列表.....	57	5.5.1 局部变量.....	78
4.1.6 列表常用内置函数和方法.....	57	5.5.2 全局变量.....	79
4.2 元组.....	58	5.6 Python 常用内置函数.....	80
4.2.1 访问元组元素.....	58	小结.....	81
4.2.2 修改元组元素.....	59	习题.....	81
4.2.3 删除元组.....	59	第 6 章 模块.....	83
4.2.4 元组运算符.....	59	6.1 模块的概念.....	83
4.2.5 元组索引、截取.....	60	6.2 模块的导入.....	83
4.2.6 无关闭分隔符.....	60	6.3 模块的发布.....	85
4.2.7 元组常用内置函数.....	60	6.4 模块的安装.....	86
4.3 字典.....	61	6.5 Python 中的标准库模块.....	86
4.3.1 访问字典元素.....	61	小结.....	87
4.3.2 修改字典.....	62	习题.....	87
4.3.3 删除字典元素.....	62	第 7 章 文件操作.....	88
4.3.4 字典键的特性.....	62	7.1 文件的定义.....	88
4.3.5 字典常用内置函数和 方法.....	63	7.2 文件的打开和关闭.....	88
4.4 集合.....	64	7.2.1 文件打开.....	88
4.4.1 可变集合.....	64	7.2.2 文件关闭.....	90
4.4.2 冻结集合.....	67	7.3 文件的写入和读取.....	91
4.4.3 集合的操作.....	71	7.3.1 写入文件.....	91
小结.....	73	7.3.2 读取文件.....	92
习题.....	73	7.4 其他操作.....	93
		7.4.1 文件定位.....	93

7.4.2 重命名和删除文件.....	95	习题.....	115
7.4.3 目录操作.....	95	第 10 章 常用扩展库.....	117
小结.....	96	10.1 NumPy 简介.....	117
习题.....	97	10.1.1 NumPy 数组.....	117
第 8 章 异常处理.....	98	10.1.2 NumPy 基本运算.....	121
8.1 异常的类型.....	98	10.2 SciPy 简介.....	124
8.2 异常的捕获及处理.....	100	10.3 Pandas 简介.....	127
小结.....	106	10.3.1 Series.....	128
习题.....	106	10.3.2 DataFrame.....	131
第 9 章 面向对象程序设计.....	107	10.4 Matplotlib 简介.....	133
9.1 面向对象的基本概念.....	107	10.4.1 快速绘图.....	133
9.2 类的声明及对象的创建.....	108	10.4.2 绘制其他图形.....	135
9.2.1 声明类.....	108	10.5 Scikit-learn 简介.....	137
9.2.2 创建对象.....	108	小结.....	141
9.2.3 访问限制.....	109	习题.....	141
9.3 封装.....	110	附录 A 常用标准库模块.....	142
9.4 继承.....	112	附录 B Python 内置函数.....	150
9.5 多态.....	114	参考文献.....	151
小结.....	115		

第 1 章 Python 概述

Python 是一种简单但功能强大的面向对象编程语言，像 Perl 语言一样，Python 源代码遵循 GPL (GNU General Public License, GNU 公共许可证) 协议。Python 语言以其优雅、简明的语法特点，使编程初学者从语法细节中摆脱出来，只需专注于要解决的问题，分析程序本身的逻辑和算法。Python 拥有大量的第三方模块，使其可以拓展到很多领域。

本章主要介绍 Python 语言的发展历史、应用领域、特点以及开发环境搭建。

1.1 Python 简史

Python 由荷兰的 Guido van Rossum 发明。Guido 在荷兰数学和计算机研究所 (CWI) 工作时，

曾参加设计过一种专门为非专业程序员 (如：数学家、物理学家等) 设计的语言——ABC。ABC 语言以教学为目的，其主要设计理念是希望让编程语言变得容易阅读、使用、记忆和学习，并以此来激发人们学习编程的兴趣。就 Guido 本人看来，ABC 这种语言非常优美和强大，但是 ABC 语言并没有成功，究其原因，Guido 认为是其非开放性造成的。因此，他想开发一种新的开源的程序设计语言。1989 年圣诞节期间，Guido 决定在继承 ABC 的基础上开发一个新的基于互联网社区的脚本解释语言，



并以其所钟爱的喜剧团体 Monty Python 将其命名为 Python。1991 年，Python 发布了第一个公开发行版。

Python 的设计理念是优美、简单、易学、易用，再加上 Python 是开源的，因此越来越多的人加入到 Python 的开发和使用当中，其功能也越来越完善。Python 不但可以用于 Web 页面的开发、网络爬虫，还逐渐成为从事数据分析、机器学习和人工智能方面的研究人员和工程技术人员的重要工具。



Python 自诞生以来不断完善和发展，应用越来越广泛。截至 2018 年 3 月，Python 在 TIOBE 编程语言排行榜上，已经上升到第四名 (见表 1-1)，稳居前五，由此可以看出 Python 的受欢迎程度。

表 1-1 TIOBE 编程语言排行榜 (TOP20)

2018 年 3 月	2017 年 3 月	名次变化	程序语言	应用百分数	变化
1	1		Java	14.941%	-1.44%
2	2		C	12.760%	+5.02%
3	3		C++	6.452%	+1.27%
4	5	^	Python	5.869%	+1.95%
5	4	v	C#	5.067%	+0.66%
6	6		Visual Basic .NET	4.085%	+0.91%
7	7		PHP	4.010%	+1.00%
8	8		JavaScript	3.916%	+1.25%
9	12	^	Ruby	2.744%	+0.49%
10	-	^^	SQL	2.686%	+2.69%
11	11		Perl	2.233%	-0.03%
12	10	v	Swift	2.143%	-0.13%
13	9	vv	Delphi/Object Pascal	1.792%	-0.75%
14	16	^	Objective-C	1.774%	-0.22%
15	15		Visual Basic	1.741%	-0.27%
16	13	v	Assembly language	1.707%	-0.53%
17	17		Go	1.444%	-0.54%
18	18		MATLAB	1.408%	-0.45%
19	19		PL/SQL	1.327%	-0.34%
20	14	vv	R	1.128%	-0.89%

1.2 Python 的应用领域

Python 简洁的语法和对动态输入的支持,再加上其解释性语言的本质,使得它在大多数平台上都是一个理想的脚本语言,特别适用于快速应用程序开发。当前,Python 已被广泛应用于众多领域,例如:

- (1) 科学运算: Python 提供了一些支持科学计算和数值分析的模块,如 NumPy、SciPy、Matplotlib、Pandas 等。
- (2) 数据分析: 2016 年 2 月 11 日,美国科学家宣布发现引力波,分析引力波数据用到了

Python 包 GWPY。

(3) 机器学习: Python 在机器学习方面一个非常强大的模块是 scikit-learn, 它是在 NumPy、SciPy 和 matplotlib 三个模块上编写的, 是数据挖掘和数据分析的一个简单而有效的工具。

(4) 云计算: 典型应用——Python 开发的 OpenStack。

(5) Web 开发: 可用于开发众多优秀的 Web 框架, 如 Django、Flask、Tornado 等。

(6) 系统运维: 开发运维人员必备的工具, 如 slatstack (系统自动化配置和管理工具)、Ansible (自动化运维工具)。

(7) 图形开发: 可用于开发 wxPython、PyQT、TKinter。

Python 在商业、艺术、科学等很多领域都有成功案例。同时, 越来越多的公司甚至政府部门将 Python 作为其主要开发语言。例如:

(1) NASA (美国宇航局): 从 1994 年起把 Python 作为主要开发语言。

(2) Dropbox (美国最大的在线云存储网站): 全部用 Python 实现, 每天网站处理 10 亿个文件的上传和下载。

(3) 豆瓣网: 图书、唱片、电影等文化产品的资料数据库网站。

(4) BitTorrent: BT 下载软件客户端。

(5) gedit: Linux 平台的文本编辑器。

(6) GIMP: Linux 平台的图像处理软件 (Linux 下的 Photoshop)。

(7) 知乎: 社交问答网站, 国内最大的问答社区, 通过 Python 开发。

(8) Autodesk Maya: 3D 建模软件, 支持 Python 作为脚本语言。

(9) YouTube: 世界上最大的视频网站 YouTube 就是用 Python 开发的。

(10) Facebook: 大量的基础库均通过 Python 实现。

(11) Redhat: 世界上最流行的 Linux 发行版本中的 yum 包管理工具使用 Python 开发。

此外, 搜狐、金山、腾讯、盛大、网易、百度、阿里、淘宝、土豆、新浪、果壳等公司都在使用 Python 完成各种各样的任务。

1.3 Python 的特点

Python 入门容易, 可从零开始学习编程。其代码可读性强, 默认情况下, 每一级缩进都是 4 个空格。Python 语言编写的程序不需要编译成二进制代码, 可以直接从源代码运行程序。在计算机内部, Python 解释器把源代码转换成字节码的中间形式, 然后再把它翻译成计算机使用的机器语言并运行。Python 也被称为是一种“胶水语言”, 可以方便地调用其他语言编写的功能模块, 并将它们有机地结合在一起形成更高效的新程序。

具体来说, Python 具有以下优点:

(1) 开源：因为 Python 遵循开源协议，所以开发人员可自由地发布 Python 文件的副本、阅读它的源代码、对它进行改动、把它的一部分用到新的自由软件中。

(2) 易学易用：Python 的定位是“优雅”“简单”“明确”，所以 Python 程序看上去总是简单易懂。初学者学 Python，不但入门容易，而且上手快，可以轻易编写复杂的程序。

(3) 开发效率高：Python 具有非常强大的第三方库，基本上可通过计算机实现任何功能。Python 官方库中有相应的模块进行支持，直接下载调用后，可在基础库的基础上再进行开发，从而大大降低开发周期，有效提高开发效率。

(4) 易移植性：由于 Python 的开源本质，Python 能够工作在不同操作系统上。如果能够避免使用依赖于操作系统的某些特性，那么几乎所有的 Python 程序无须修改就可以在不同的操作系统上运行。

当然，同其他语言一样，Python 也有一些自身的缺点，其主要的缺点就是运行速度不够快，程序运行的效率不如 Java 或者 C 语言高。但这里所指的速度慢在大多数时候用户是无法感知的，例如，使用 C 语言程序开发一个程序，执行时间需要 0.01 s；使用 Python 实现同样功能的程序，需要花费 0.1 s 的时间，虽然相差 10 倍，但用户是感觉不到的。

1.4 Python 安装及开发环境构建

1.4.1 Python 3.6.5 下载与安装

Python 的下载地址是 <https://www.python.org/downloads/>，该页面如图 1-1 所示。这里有两个版本的 Python 可供下载，一个版本是 V3.6.5，另一个是 V2.7.14，用户可以根据需要进行下载。需要注意的是 V2.7.14 版本的 Python 将在 2020 年后不再更新，因此建议用户下载 V3.6.5 版本的 Python。本书以 Python 3.6.5 为版本进行讲解。

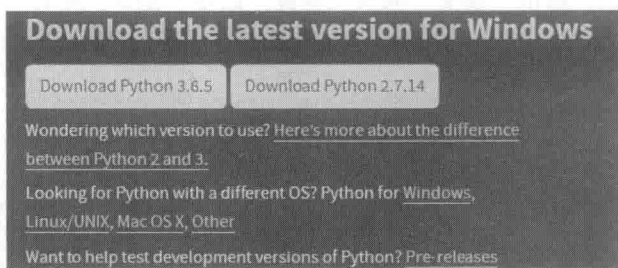


图 1-1 Python 下载页面

这里提供 32 位和 64 位的 Python，用户可以根据计算机操作系统的位数选择不同 Python 安装文件，下载之后就可以双击该文件进入安装界面，如图 1-2 所示。在该页面下，可以选择默认安装 Install Now，也可以选择自定义安装 Customize installation，自定义安装更加灵活一些，

用户可以根据需要设置安装参数。同时,用户可以选中 Add Python3.6 to PATH 复选框,将 Python 自动加入到系统的环境变量中,从而避免安装后再自己设置 Python 的环境变量。



图 1-2 Python 安装界面

如果选择 Customize installation, 将进入 Optional Features (可选组件) 安装界面 (见图 1-3), 用户可以自行选择需要安装的组件, 然后单击 Next 按钮进入 Advanced Options (高级选项) 设置界面 (见图 1-4), 用户可以设置 Python 在本机的安装路径。在选择完需要的组件并设置安装路径后, 单击 Install 按钮就可以执行 Python 的安装过程, 如图 1-5 所示。当出现如图 1-6 所示的界面时, 即表示 Python 已经完成在本机的安装过程。

在 cmd 窗口下输入 python -V 命令, 显示 Python 3.6.5, 这表明系统中已经成功安装了 Python 环境, 如图 1-7 所示。

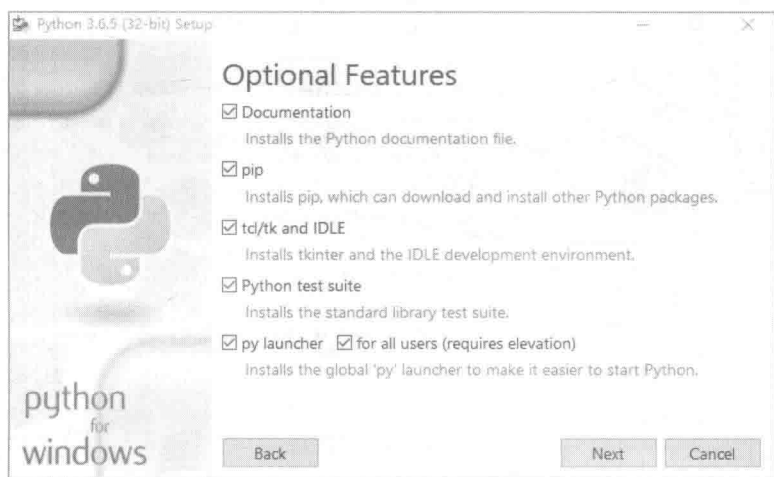


图 1-3 可选组件界面

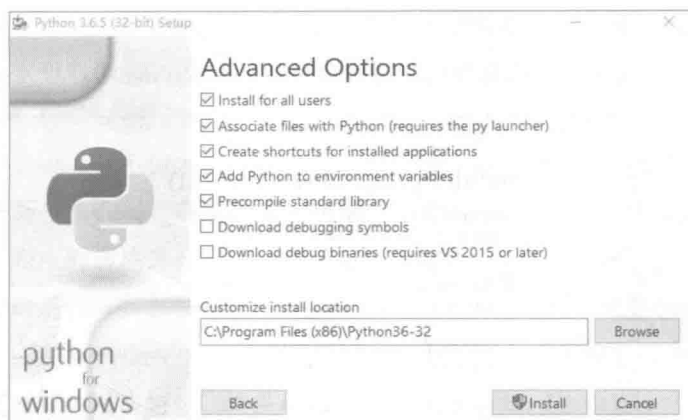


图 1-4 高级选项界面

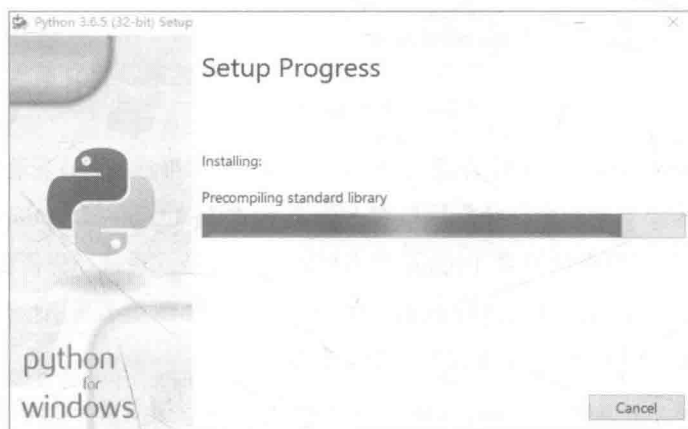


图 1-5 Python 安装进程界面



图 1-6 Python 安装成功界面



图 1-7 cmd 界面

安装完 Python 后，通过选择 IDLE，可以输入 shell 命令（见图 1-8），用户可以输入第一条 Python 命令：`print('Hello World!')`，将会输出 `Hello World!`（见图 1-9）。

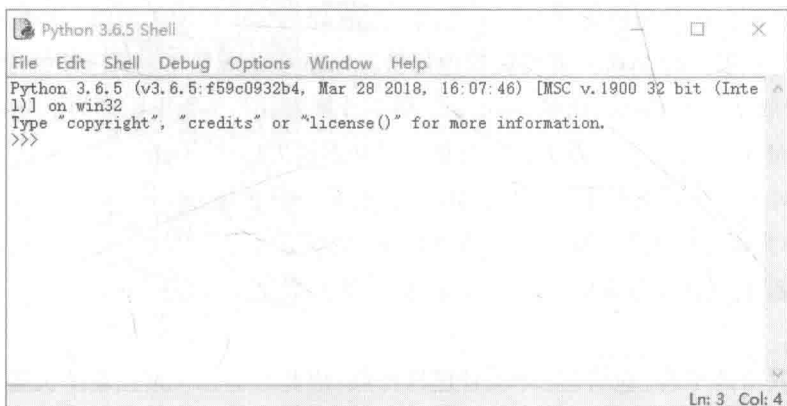


图 1-8 Python 的 Shell 界面

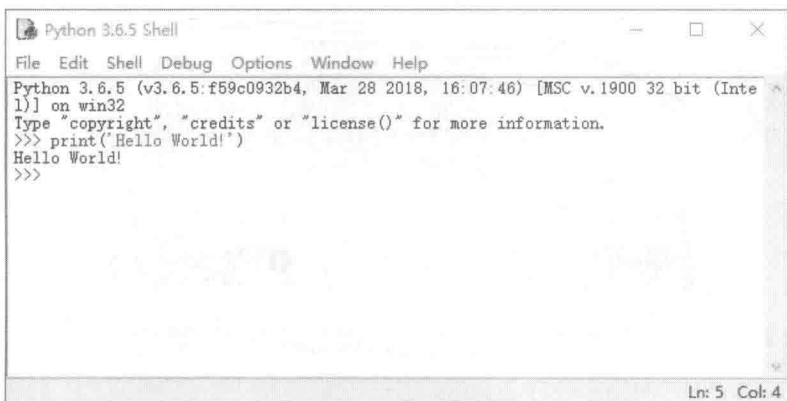


图 1-9 Python 的 Hello World

1.4.2 Anaconda 下载与安装

Python 易用，但用好却不容易，例如，包管理和 Python 不同版本的问题。有时为了安装不同版本的 Python，需要下载不同的版本。因为 Python2.x 版本与 3.x 版本有些语法不兼容，因此当需要特定版本的 Python 时需要进行环境选择。

为了解决这些问题，有不少发行版的 Python（如 Anaconda 等）。其将许多常用的 package 打包，从而方便用户直接使用。Anaconda 一般集成了很多常用的 package，用户不需要再重复下载、安装、配置环境变量等烦琐的操作就可以直接使用。

Anaconda 主要是一个用于科学计算的 Python 发行版，支持 Linux、Mac、Windows 等多种系统，提供了包管理与环境管理的功能，可以很方便地解决多版本 Python 并存、切换，以及各种第三方包安装问题。Anaconda 利用工具/命令 conda 来进行 package 和 environment 的管理，并且已经包含了 Python 和相关的配套工具。

这里先解释一下 conda、anaconda 这些概念的差别。conda 可以理解为一个工具，也是一个可执行命令，其核心功能是包管理与环境管理。包管理与 pip 的使用类似，环境管理则允许用户方便地安装不同版本的 Python 并可以快速切换。Anaconda 则是一个打包的集合，里面预装好了 conda、某个版本的 Python、众多 package、科学计算工具等，所以也称为 Python 的一种发行版。其实还有 Miniconda，顾名思义，它只包含最基本的内容——Python 与 conda，以及相关的必须依赖项，对于空间要求严格的用户，Miniconda 是一种选择。

conda 几乎将所有的工具、第三方包都当作 package 对待，甚至包括 Python 和 conda 自身，因此，conda 打破了包管理与环境管理的约束，能非常方便地安装各种版本 Python、各种 package 并方便地切换。

Anaconda 具有跨平台、包管理、环境管理的特点，因此很适合快速在新的机器上部署 Python 环境。为了下载 Anaconda，用户需要首先登录到 <https://www.anaconda.com/download/>，如图 1-10 所示。



图 1-10 Anaconda 下载界面

2018年2月发布的V5.1是Anaconda的较新版本,集成的Python版本有两个:V3.6和V2.7,因为Python 2.7将于2020年不再支持,因此推荐下载集成Python 3.6版本的Anaconda,默认的操作系统是Windows。根据安装计算机的位数可以选择64位或者32位安装文件。单击Anaconda安装文件,在安装界面(见图1-11)下直接单击Next按钮进入到Anaconda的安装协议界面(见图1-12),在同意安装协议后将进入到选择安装类型界面(见图1-13),可以选择只能登录用户使用或者本机的所有用户使用两种类型。单击Next按钮进入选择安装路径界面(见图1-14),用户可以自行选择Anaconda在本机的安装路径。在高级安装选项界面(见图1-15)下,用户可以选择自动设置Anaconda的环境变量和将Anaconda提供的Python版本作为默认的Python环境,然后单击Install按钮开始执行Anaconda的安装(见图1-16和图1-17),图1-18所示为Anaconda在本机已经成功完成安装。在本机Anaconda的安装路径下选择Anaconda Navigator将会展示已安装的Anaconda组件,如图1-19所示。

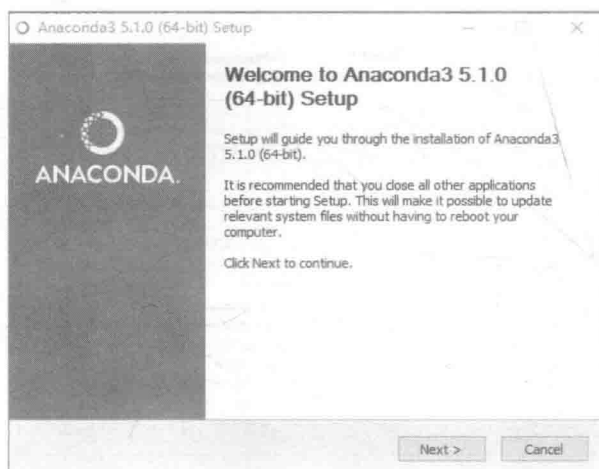


图 1-11 Anaconda 安装界面

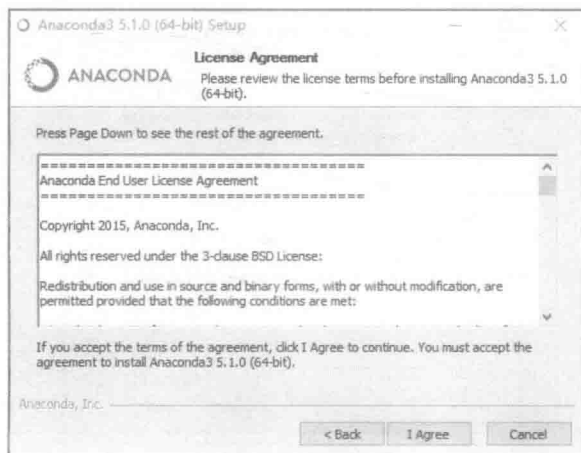


图 1-12 Anaconda 安装协议界面