

普通高等教育新工科人才培养规划教材

大数据专业

Python

程序设计教程

主 编 李治国 武春岭
副主编 唐乾林 梁雪梅 鲁先志 周璐璐 赵 怡



普通高等教育新工科人才培养规划教材（大数据专业）

Python 程序设计教程

主 编 李治国 武春岭

副主编 唐乾林 梁雪梅 鲁先志 周璐璐 赵 怡



中国水利水电出版社
www.waterpub.com.cn

· 北京 ·

内 容 提 要

目前已经出版的 Python 相关教材大多以 Python 2.0 为平台编写,具有一定的局限性。其程序代码已经无法直接在最新的 Python 平台上运行,不能很好地满足读者对 Python 语言的学习需求。本书以 Python 3.0 为基础编写,融入了最新的 Python 语言和编程特点。本书一共 11 章,从基本语法入手,涵盖了 Python 语言中的常见序列结构、常见语句、函数文件、类和继承、多线程编程、图形界面设计、数据库应用、网络应用和 Web 应用等内容。本书采用案例引导的方式,每个章节精心编排了大量的案例程序,生动形象地向学生展示了知识结构和项目应用。书中的所有程序都经过调试运行,保证了案例程序的正确性。

本书具有清晰易懂、案例丰富、实战性强的特点,适合本科和高职高专学生作为学习教程,同时也可以作为 Python 编程爱好者和程序员的学习和参考资料。

图书在版编目(CIP)数据

Python 程序设计教程 / 李治国, 武春岭主编. -- 北京: 中国水利水电出版社, 2018.7
普通高等教育新工科人才培养规划教材. 大数据专业
ISBN 978-7-5170-6588-3

I. ①P… II. ①李… ②武… 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138025号

策划编辑: 寇文杰 责任编辑: 张玉玲 加工编辑: 张青月 封面设计: 李 佳

书 名	普通高等教育新工科人才培养规划教材(大数据专业) Python 程序设计教程 Python CHENGXU SHEJI JIAOCHENG
作 者	主 编 李治国 武春岭 副主编 唐乾林 梁雪梅 鲁先志 周璐璐 赵 怡
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路1号D座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 68367658 (营销中心)、82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	三河市鑫金马印装有限公司
规 格	184mm×260mm 16开本 12印张 295千字
版 次	2018年7月第1版 2018年7月第1次印刷
印 数	0001—3000册
定 价	34.00元



凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换
版权所有·侵权必究

前 言

Python 语言是一种计算机编程语言,作用类似于 C/C++/Java/Perl/VB/Delphi 等计算机编程语言,具有非常清晰易读的语法特点,是一种面向对象的高级语言,并且可以进行扩展。Python 语言用途非常广泛,支持 Java 和 .Net 技术,可以运行在 Windows、Linux、FreeBSD、Solaris 等几乎所有的操作系统上,也可以运行在手机中。目前在国际上非常流行,正在得到越来越多的重视。

Python 语言使用方便,不需要进行复杂的编译,可以进行各种软件的开发,比如:制作网站、开发图形界面(GUI)程序、网络编程、数据库编程、图形图像处理、科学计算、手机编程、游戏编程等。

本书基于 Python 3.0 版本编写。Python 3.0 是目前 Python 的较新版本,相比之前的版本在部分语法上更加精炼合理,并且得到更多第三方软件的支持,拥有更加广阔的资源。由于 Python 3.0 版本在设计时没有考虑向下兼容,因此许多早期 Python 版本设计的程序都无法在 Python 3.0 上正常执行。本教程中的所有案例均需要在 Python 3.0 环境下运行和调试。

本教程一共 11 章,从基本语法入手,循序渐进,从理论延伸到实践,将读者逐步引入到 Python 程序设计的精彩世界中。第 1 章介绍了 Python 语言的特点,该语言的编译器及开发环境的安装方法以及在编程过程中的程序调试环境。第 2 章主要讲解 Python 语言的变量类型和常用语句。第 3 章阐述了字符串、列表、元组、集合、字典等五种数据结构和应用。第 4 章介绍函数的概念、函数的定义方法以及如何调用函数。第 5 章介绍利用 Python 语言中的输入和输出功能、读取和写入的方法,以及文件内建函数、方法、属性及文件系统等内容。第 6 章讲解 Python 语言的面向对象编程。第 7 章介绍基于 Tkinter 模块的图形界面编程。第 8 章阐述线程创建、线程同步和线程优先级等内容。第 9 章基于 SQLite 数据库和 MySQL 数据库,详细介绍数据库的创建、查询和修改等内容。第 10 章讲解网络中最常用的套接字和邮件服务等网络编程和应用。第 11 章讲述 Web 开发,介绍与 Python Web 开发技术相关的 WSGI 框架和模板的使用。

全书由重庆电子工程职业学院的李治国、武春岭任主编,唐乾林、梁雪梅、鲁先志、周璐璐、赵怡为副主编。中国水利水电出版社的寇文杰编辑对本书的出版给予了大力支持。在此,谨向为本书出版付出辛勤劳动的同志表示感谢。

由于编者水平有限,书中不足之处和错误在所难免,恳请广大读者批评指正,我们将在再版时及时改进。编者的 E-mail: 578774623@qq.com。

编 者

2018 年 4 月

目 录

前言	
第1章 Python 概述	1
1.1 Python 语言概述	1
1.1.1 什么是 Python 语言	1
1.1.2 Python 语言特点	2
1.2 Python 开发环境的安装与配置	2
1.2.1 Python 安装	3
1.2.2 环境变量配置	3
1.3 IDLE 编程环境	4
1.3.1 通过交互模式进行编程	5
1.3.2 通过脚本模式进行编程	5
1.3.3 使用 IDLE 的调试器	6
1.4 PyCharm 编程环境	7
习题	10
第2章 Python 程序设计基础	12
2.1 Python 基本语法	12
2.1.1 标识符	12
2.1.2 程序注释	12
2.1.3 代码块和组	13
2.1.4 基本输出语句	14
2.2 变量和数字类型	14
2.2.1 变量	14
2.2.2 数据类型	15
2.2.3 数字类型转换	15
2.3 使用解释器	16
2.3.1 交互式编程	16
2.3.2 脚本式编程	16
2.4 运算符和优先级	17
2.4.1 运算符	17
2.4.2 优先级	21
2.5 条件控制语句	21
2.5.1 if 语句	22
2.5.2 if 嵌套	23
2.6 循环语句	24
2.6.1 while 语句	24
2.6.2 for 语句	26
2.6.3 break 和 continue 语句	28
2.7 迭代器和生成器	31
2.7.1 迭代器	31
2.7.2 生成器	32
习题	32
第3章 序列数据结构	34
3.1 字符串	34
3.1.1 字符串查询	34
3.1.2 字符串更新	35
3.1.3 转义字符	35
3.1.4 字符串运算符	36
3.1.5 字符串格式化	38
3.2 列表	38
3.2.1 列表赋值	39
3.2.2 列表查询	39
3.2.3 列表更新	39
3.2.4 列表元素删除	40
3.2.5 列表操作符	41
3.2.6 列表嵌套	42
3.3 元组	42
3.3.1 元组查询	43
3.3.2 元组修改	43
3.3.3 删除元组	43
3.3.4 元组运算符	44
3.4 集合	45
3.4.1 集合创建	45
3.4.2 集合运算	45
3.5 字典	46
3.5.1 字典查询	47
3.5.2 字典修改	48
3.5.3 字典元素删除	48

3.5.4 字典的特性	48	5.3.5 seek()函数	75
习题	49	5.3.6 tell()函数	76
第4章 函数和模块	51	习题	77
4.1 函数	51	第6章 面向对象编程	78
4.1.1 函数定义	51	6.1 创建类	78
4.1.2 函数调用	52	6.1.1 类的定义	78
4.2 参数传递	54	6.1.2 类的实例化	78
4.2.1 参数传递对象	54	6.1.3 类的方法	79
4.2.2 参数传递类型	55	6.1.4 构造方法	80
4.3 匿名函数	58	6.1.5 私有属性和方法	80
4.4 返回值	58	6.2 继承	82
4.5 变量作用域	59	6.2.1 继承的定义和特征	82
4.5.1 作用域的范围	59	6.2.2 单继承	83
4.5.2 全局变量和局部变量	59	6.2.3 多继承	84
4.5.3 global 和 nonlocal 关键字	60	6.2.4 方法重写	85
4.6 模块	61	6.2.5 运算符重载	86
4.6.1 模块定义	61	习题	88
4.6.2 模块导入	62	第7章 GUI编程	90
4.7 标准模块	63	7.1 Tkinter 模块功能	90
4.8 时间模块	63	7.1.1 创建一个GUI程序	90
4.8.1 时间戳	63	7.1.2 Tkinter 控件简介	91
4.8.2 获取当前时间	64	7.2 Tkinter 图形界面控件	92
4.8.3 获取格式化时间	64	7.2.1 Label 控件	92
4.8.4 格式化日期	64	7.2.2 Button 控件	93
4.8.5 获取某月日历	66	7.2.3 Canvas 控件	94
习题	66	7.2.4 Checkbutton 控件	95
第5章 输入输出和文件	68	7.2.5 Radiobutton 控件	98
5.1 输入输出	68	7.2.6 Entry 控件	101
5.1.1 输出格式	68	7.2.7 Combobox 控件	102
5.1.2 键盘输入	70	7.2.8 ScrolledText 控件	104
5.2 文件操作	70	7.2.9 Menu 控件	108
5.2.1 open()函数	70	7.2.10 Frame 控件	109
5.2.2 close()函数	71	7.3 事件响应	113
5.2.3 文件对象属性	72	7.3.1 鼠标事件	113
5.3 文件对象操作	73	7.3.2 键盘事件	115
5.3.1 read()函数	73	习题	117
5.3.2 write()函数	73	第8章 多线程编程	119
5.3.3 readline()函数	74	8.1 进程和线程简介	119
5.3.4 next()函数	74	8.1.1 进程和线程的概念	119

8.1.2	进程与线程之间的关系	119
8.2	线程创建	120
8.2.1	函数方法创建线程	120
8.2.2	用 threading 模块创建线程	121
8.3	线程同步	123
8.3.1	线程锁	123
8.3.2	threading.RLock 和 threading.Lock 的区别	126
8.3.3	BoundedSemaphore	126
8.3.4	event	128
8.3.5	conditions	130
8.3.6	barriers	132
8.4	Queue 模块	133
8.4.1	FIFO 队列	133
8.4.2	LIFO 队列	133
	习题	137
第 9 章	数据库编程	139
9.1	数据库简介	139
9.1.1	数据库系统管理	139
9.1.2	关系型数据库	140
9.2	SQLite 数据库应用	141
9.2.1	关于 SQLite 数据库	141
9.2.2	连接 SQLite 数据库	141
9.2.3	创建表	142
9.2.4	删除表	144
9.2.5	向表中添加数据	144
9.2.6	查找数据	145
9.2.7	更新数据	146
9.2.8	删除数据	148
9.3	MySQL 数据库应用	149
9.3.1	关于 MySQL 数据库	149
9.3.2	安装 MySQL 数据库	149
9.3.3	安装 PyMySQL 模块	151
9.3.4	连接数据库	152
9.3.5	创建表	152
9.3.6	插入数据	153
9.3.7	查询数据	154
9.3.8	更新数据	155
9.3.9	删除数据	156
	习题	156
第 10 章	网络编程应用	158
10.1	Socket 编程	158
10.1.1	套接字模块	158
10.1.2	编写一个简单的服务器	160
10.2	邮件服务程序	161
10.2.1	发送普通电子邮件	162
10.2.2	发送 HTML 电子邮件	163
10.2.3	发送带附件的电子邮件	164
10.2.4	在 HTML 文本中添加图片	165
	习题	166
第 11 章	Web 开发	168
11.1	Web 服务简介	168
11.1.1	HTTP 协议	169
11.1.2	HTTP 跟踪	170
11.1.3	HTTP 格式	170
11.2	超文本	171
11.2.1	HTML	171
11.2.2	CSS	172
11.2.3	JavaScript	174
11.3	WSGI 接口	175
11.3.1	WSGI 接口介绍	175
11.3.2	运行 WSGI 服务	176
11.4	Web 框架	178
11.4.1	Flask 框架简介	178
11.4.2	Flask 框架应用	178
11.5	模板	181
11.5.1	模板的功能	181
11.5.2	MVC 框架	181
11.5.3	MVC 应用	182
	习题	184
	参考文献	185

第 1 章 Python 概述

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。Python 由 Guido van Rossum 于 1989 年底发明，第一个公开发行人版发行于 1991 年。由于 Python 语言的简洁性、易读性以及可扩展性，它目前已经成为一种主流的、最受欢迎的程序开发语言之一。

本章学习重点：

- Python 语言的特点
- Python 3.0 安装和环境配置方法
- IDLE 常用功能
- IDLE 程序调试环境
- PyCharm 编程环境

1.1 Python 语言概述

1.1.1 什么是 Python 语言

Python 语言是一种面向对象的用途非常广泛的编程语言，具有非常清晰的语法特点，适用于多种操作系统，可以在 Windows 和 UNIX 这样的系统中运行。目前在国际上非常流行，正在得到越来越多的应用。Python 可以完成许多任务，功能非常强大。Python 的官方网站是：<http://www.python.org/>，可以在该网站找到很多相关资料。

Python 语言使用方便，不需要进行复杂的编译，用途非常广泛，可以进行各种软件的开发，比如：制作网站、开发图形界面（GUI）程序、网络编程、数据库编程、图形图像处理、科学计算、手机编程等。使用 Python 最多的应该是 Google 公司了，Google 搜索引擎就是该公司的产品。微软公司也已经开始提供 Python 语言的软件了。全球著名的手机厂商 Nokia 公司早已开始提供基于 Python 语言的手机开发软件了。另外，还有很多游戏是用 Python 开发的。目前 Python 已经有成百上千的公共资源供用户使用。

在国外，用 Python 做科学计算的研究机构日益增多，一些知名大学已经采用 Python 来教授程序设计课程。例如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。众多开源的科学计算软件包都提供了 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。Python 专用的科学计算扩展库就更多了，例如三个十分经典的科学计算扩展库：NumPy、SciPy 和 matplotlib。它们分别为 Python 提供了快速数据处理、数值运算以及绘图功能。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和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 语言特点

Python 是结合了解释性、编译性、互动性和面向对象的高层次的脚本语言。Python 的设计具有很强的可读性，相比其他语言经常使用英文关键字，它更有特色。

(1) Python 语言具有以下结构特点：

- Python 是解释型语言。这意味着开发过程中没有了编译这个环节，类似于 PHP 和 Perl 语言。
- Python 是交互式语言。这意味着可以在一个 Python 提示符下，直接互动执行你写的程序。
- Python 是面向对象语言。这意味着 Python 支持面向对象的风格或代码封装在对象的编程技术。
- Python 是初学者的语言。对初级程序员而言，Python 是一种伟大的语言，它支持广泛的应用程序开发，从简单的文字处理到 WWW 浏览器，再到游戏。

(2) Python 语言具有以下兼容性特点：

- 易于学习：Python 有相对较少的关键字和明确定义的语法，结构简单，学习起来更加容易。
- 易于阅读：Python 代码定义更清晰。
- 易于维护：Python 的源代码是相当容易维护的。
- 广泛的标准库：Python 的最大的优势之一是具有丰富的、跨平台的库，在 UNIX、Windows 和 Macintosh 系统的兼容性很好。
- 互动模式：互动模式支持从终端输入执行代码并获得结果，并且支持互动测试和调试代码片断。
- 可移植：基于其开放源代码的特性，Python 已经被移植（也就是使其工作）到许多平台。
- 可扩展：如果需要一段运行很快的关键代码，或者是想要编写一些不愿开放的算法，可以使用 C 或 C++ 完成那部分程序，然后从 Python 程序中调用。
- 数据库：Python 提供所有主要的商业数据库的接口。
- GUI 编程：Python 支持将创建的 GUI 移植到许多系统调用。
- 可嵌入：可以将 Python 嵌入到 C/C++ 程序，程序的使用者可获得“脚本化”的能力。

1.2 Python 开发环境的安装与配置

除了应用于 Windows 平台，Python 还可应用于多种平台，包括 Linux 和 Mac OS X。通过在终端窗口输入“python”命令，可查看本地是否已经安装了 Python 及其安装版本，如图 1-1 所示。

Python 的最新源码、二进制文档、新闻资讯等可以在 Python 的官网查看，Python 官网是 <http://www.python.org/>。还可以在文档下载地址 www.python.org/doc/ 中下载 Python 的文档，包括 HTML（超文本标记语言）、PDF 和 PostScript 等格式的文档。



```
命令提示符 - python
Microsoft Windows [版本 10.0.15063]
(c) 2017 Microsoft Corporation. 保留所有权利。

C:\Users\57877>python
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
```

图 1-1 查看 Python 的安装版本

1.2.1 Python 安装

Python 已经被移植到许多不同平台，在不同平台上 Python 需要下载各相应平台的二进制代码，然后再进行安装。

1. 在 UNIX & Linux 平台安装 Python

以下为在 UNIX & Linux 平台上安装 Python 的简单步骤：

- (1) 打开 Web 浏览器访问 <http://www.python.org/download/>。
- (2) 选择适用于 UNIX/Linux 的源码压缩包。
- (3) 下载及解压压缩包。
- (4) 在 Modules/Setup 内定义、修改需要的选项及设置。
- (5) 执行 ./configure 脚本。
- (6) make。
- (7) make install。

执行以上操作后，Python 会安装在 /usr/local/bin 目录中，Python 库安装在 /usr/local/lib/pythonXX，XX 是使用的 Python 的版本号。

2. 在 Windows 平台安装 Python

以下为在 Windows 平台上安装 Python 的简单步骤。

- (1) 打开 Web 浏览器访问 <http://www.python.org/download/>，在下载列表中选择 Window 平台安装包，包文件为：python-XYZ.msi，XYZ 为要安装的 Python 的版本号。
- (2) Windows 系统必须支持 Microsoft Installer 2.0 方可使用安装程序 python-XYZ.msi 进行安装。将安装文件保存到本地计算机，然后运行它。
- (3) 双击下载包，进入 Python 安装向导，安装非常简单，使用默认的设置一直单击“下一步”按钮直到安装完成即可。

3. 在 Mac 平台安装 Python

最近的 Mac 系统都自带有 Python 环境，也可以在链接 <http://www.python.org/download/> 下载最新版本进行安装。

1.2.2 环境变量配置

程序和可执行文件可能存于与多个目录，而这些路径很可能不在操作系统提供的可执行文件的搜索路径中。Path（路径）存储的环境变量名是由操作系统维护的一个字符串命名。环

境变量包含可用的命令行解释器和其他程序的信息。

在 UNIX 或 Windows 平台中路径变量为 PATH (UNIX 区分大小写, Windows 不区分大小写)。在 Mac OS 平台中, 安装程序过程中改变了 Python 的安装路径。如果需要在其他目录引用 Python, 必须在 Path 中添加 Python 目录。

1. 在 UNIX/Linux 中设置环境变量

- 在 csh shell 输入 `setenv PATH "$PATH:/usr/local/bin/python"`, 然后按 Enter 键。
- 在 bash shell (Linux) 输入 `export PATH="$PATH:/usr/local/bin/python"`, 然后按 Enter 键。
- 在 sh 或者 ksh shell 输入 `PATH="$PATH:/usr/local/bin/python"`, 然后按 Enter 键。

提示: /usr/local/bin/python 是 Python 的安装目录。

2. 在 Windows 中设置环境变量

(1) 可以在环境变量中添加 Python 目录:

在命令提示框 (cmd) 中输入 `Path=%Path%;C:\Python`, 然后按 Enter 键。

注意: C:\Python 是 Python 的安装目录。

(2) 也可以通过以下方式设置:

- 1) 右键单击“计算机”, 然后选择“属性”命令。
- 2) 单击“高级系统设置”选项。
- 3) 选择“系统变量”窗口下面的 Path, 双击即可。
- 4) 在 Path 行, 添加 Python 安装路径。

注意: 路径要用分号“;”隔开。

表 1-1 所示是 Python 环境变量描述, 它们是 Python 的几个重要环境变量。

表 1-1 Python 环境变量描述

环境变量	描述
PYTHONPATH	PYTHONPATH 是 Python 搜索路径, 默认 import 的模块都会从 PYTHONPATH 里面寻找
PYTHONSTARTUP	Python 启动后, 先寻找 PYTHONSTARTUP 环境变量, 然后执行此变量指定的文件中的代码
PYTHONCASEOK	加入环境变量 PYTHONCASEOK, 会使 Python 导入模块的时候不区分大小写
PYTHONHOME	另一种模块搜索路径, 通常内嵌于 PYTHONSTARTUP 或 PYTHONPATH 目录中, 使得两个模块库更容易切换

1.3 IDLE 编程环境

IDLE 是 Python 软件包自带的一个集成开发环境, 非常适合 Python 编程的初学者。当安装好 Python 后, IDLE 就自动安装好了, 不需要另外安装。同时, 使用 Eclipse 这个强大的框架时 IDLE 也可以非常方便地调试 Python 程序。用户可以将其看作是一个用于编程的文字处理器, 但它能做的事情不止是编写、保存、编辑那么简单。IDLE 基本功能包括: 语法高亮; 段落缩进; 基本文本编辑; Tab 键控制; 调试程序。

IDLE 提供了两种工作模式：交互模式（interactive mode）和脚本模式（script mode）。

1.3.1 通过交互模式进行编程

最简单的方式是以交互模式启动 Python。在该模式中，编程者告诉 Python 要做什么，Python 就会立即给出响应。在 Python 安装完成之后，可以在开始菜单中选择 IDLE 命令，这样就启动了一个交互式会话界面 IDLE Shell 窗口，其操作界面如图 1-2 所示。

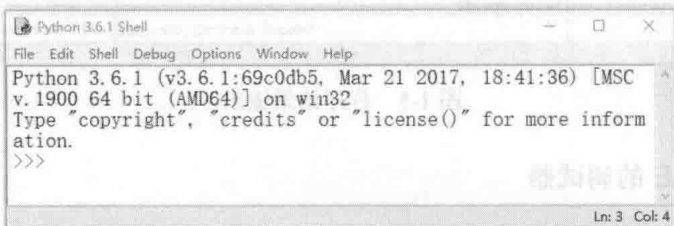


图 1-2 IDLE Shell 窗口

与很多程序语言的学习一样，我们学习的第一个 Python 程序语句是从“Hello world”开始。在命令提示符（>>>）后面输入 `print("Hello world!")` 并按下 Enter 键，解释器就会在屏幕上输出结果，如图 1-3 所示。这是我们的第一个 Python 程序的运行结果。

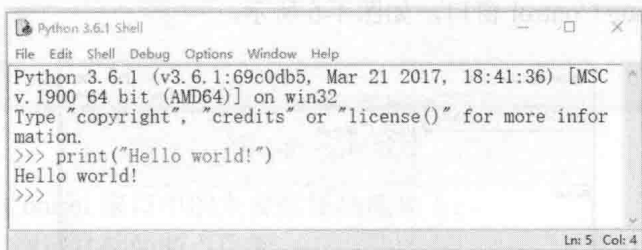


图 1-3 解释器输出结果

1.3.2 通过脚本模式进行编程

交互模式能让用户即刻得到反馈，看到结果。但如果想创建一个程序并将其保存起来以便今后还可以再执行的话，交互模式就不那么合适了。因此 Python 的 IDLE 还提供了一个脚本模式，在该模式下，可以编写、编辑、加载以及保存程序，它就好像是一个代码的文字处理器。事实上，确实可以用上一些类似的功能，比如查找和替换、剪切和粘贴等。

单击 File→NewFile 可以创建一个新的 Python 脚本文件，在该文件中编写 Python 程序，如图 1-4 所示。

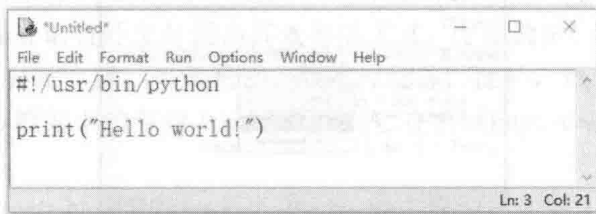
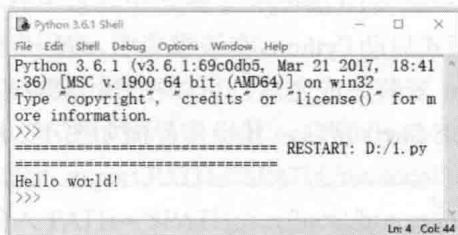


图 1-4 脚本编程界面

在 Run 菜单项的下拉菜单中选择 Run Module 命令或者按下 F5 键运行程序, 结果如图 1-5 所示。



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41
:36) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for m
ore information.
>>>
===== RESTART: D:/1.py
=====
Hello world!
>>>
```

图 1-5 程序结果输出

1.3.3 使用 IDLE 的调试器

IDLE 还为用户提供了调试器, 帮助开发人员查找逻辑错误。利用调试器可以分析被调试程序的数据, 并监视程序的执行流程。调试器的功能包括暂停程序执行、检查和修改变量、调用方法而不更改程序代码等。

使用 IDLE 的调试器进行程序调试的步骤为:

(1) 在 Debug 菜单项的下拉菜单中选择 Debugger 命令, 就启动了 IDLE 的交互式调试器, 这时 IDLE 会打开 Debug Control 窗口, 如图 1-6 所示。

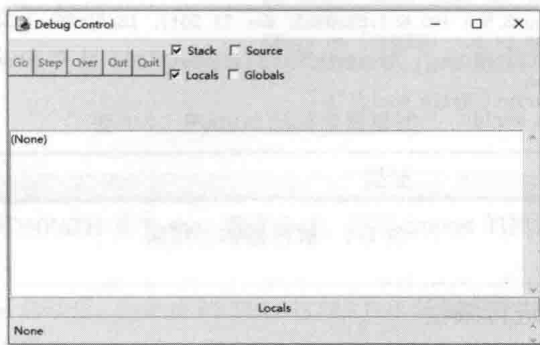


图 1-6 Debug Control 窗口

(2) 在该 Shell 中打开想要调试的 py 文件, 选中需要进行调试的代码行, 单击右键, 在弹出的快捷菜单中选择 SetBreakpoint 命令设置断点位置, 选择 Clear Breakpoint 命令可以取消断点设置, 如图 1-7 所示。

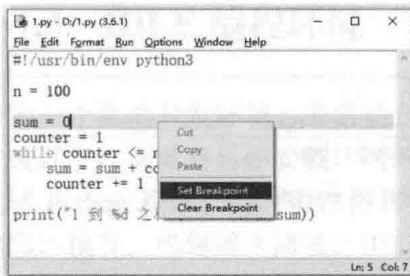


图 1-7 设置断点

(3) 在需要调试的 py 文件窗口中, 在 Run 菜单项的下拉菜单中选择 Run Module 命令或者按下 F5 键运行文件, 就可以进入调试过程, 如图 1-8 所示。

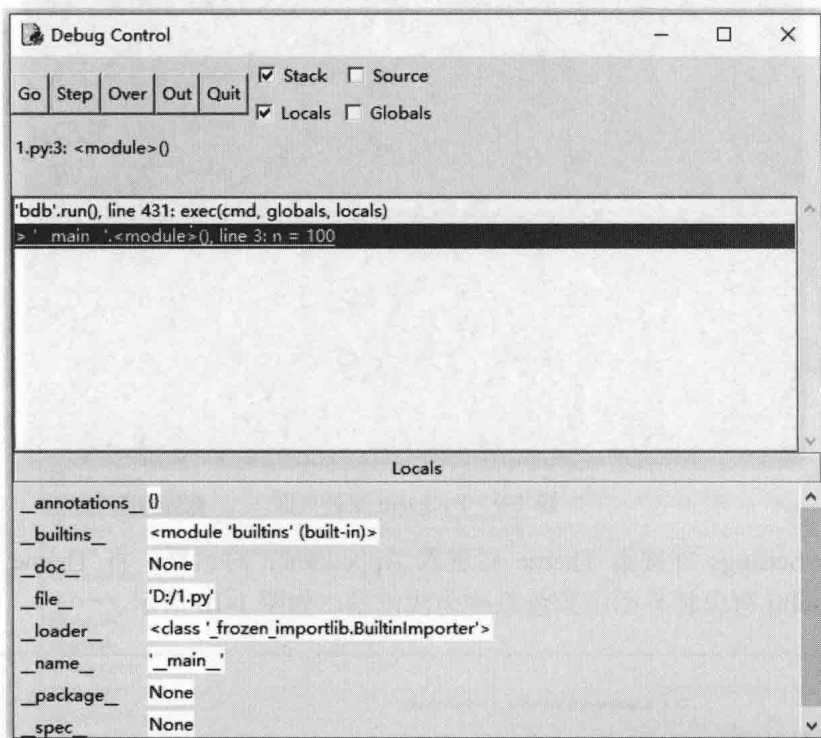


图 1-8 调试过程

在当前的 Debug Control 窗口中的主要按钮功能如下:

- Go 表示进入调试文件的断点位置。
- Step 表示调试过程中运行进入函数内部。
- Over 表示调试过程中不进入函数内部。
- Out 表示调试过程中从函数内部跳出。
- Quit 表示结束调试。

(4) 如果要退出调试器, 可以再次选择 Debug 菜单项下的 Debugger 命令, 这时 IDLE 会关闭 Debug Control 窗口并在 Python Shell 窗口中输出[DEBUG OFF]。

1.4 PyCharm 编程环境

PyCharm 是一款 Python 的 IDE 的编辑工具, 它是由 JetBrains 出品的产品, 带有一整套可以帮助用户在使用 Python 语言开发时提高其效率的工具, 比如调试、语法高亮、Project 管理、代码跳转、智能提示、自动完成、单元测试、版本控制等。此外, 该 IDE 提供了一些高级功能, 以用于支持 Django 框架下的专业 Web 开发。可以在官网<http://www.jetbrains.com/pycharm> 下载并安装 Pycharm。

安装完成后单击 Create New Project 进入 Pycharm 主界面, 如图 1-9 所示。

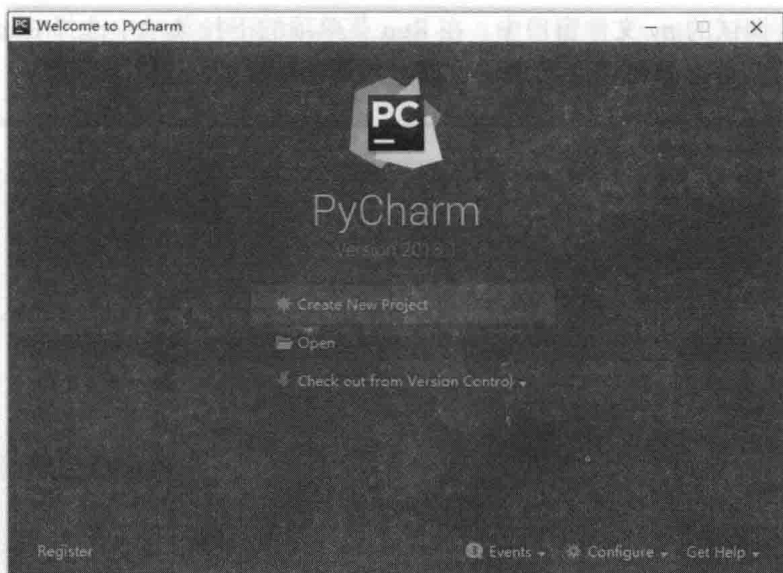


图 1-9 Pycharm 安装页面

单击 File→Settings 并搜索 Theme 后进入 Appearance 对话框，将 Theme 下拉菜单中的 Darcula 改为 IntelliJ 可以将界面由黑色系改为浅色系，如图 1-10 所示。

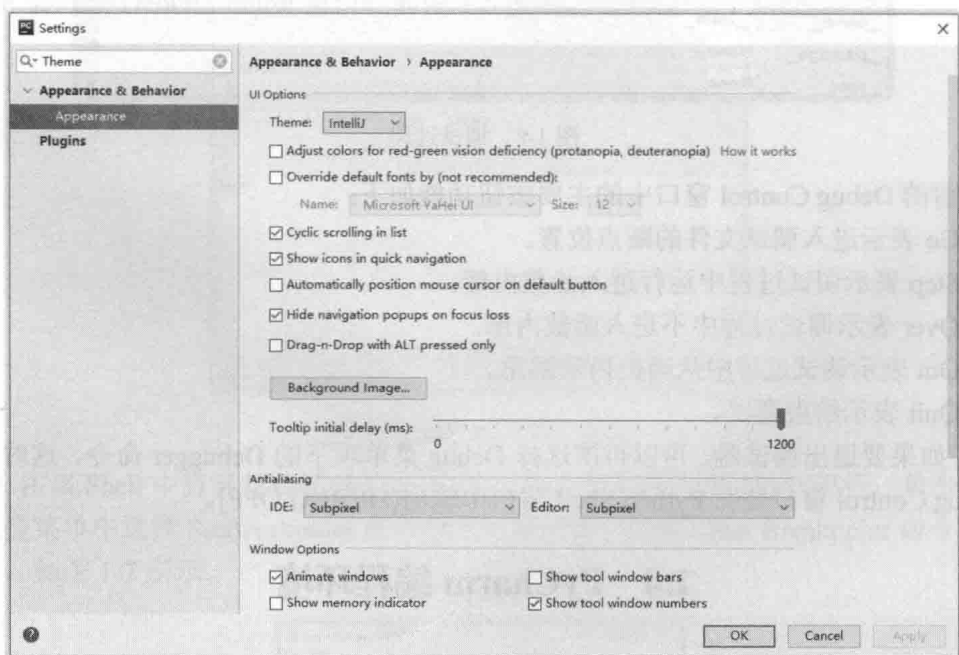


图 1-10 修改主页面色调。

单击 File→Settings→Editor→Colors Scheme→Console Font 可以更改代码的字体及其大小和行距，如图 1-11 所示。

单击 File→New Project→Pure Python 选择 Create 命令，就可以创建一个 Python 文件，如图 1-12 所示。

在代码编辑框中编辑代码，单击 Run 按钮运行代码，如图 1-13 所示。

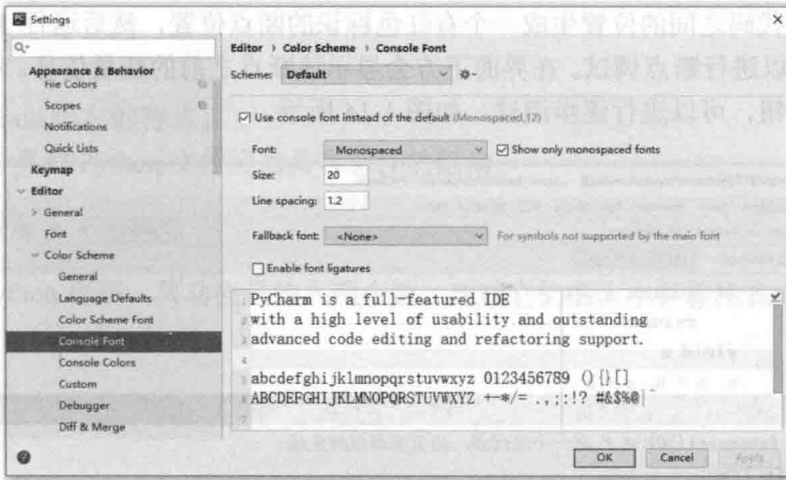


图 1-11 更改代码的字体及其大小和行距

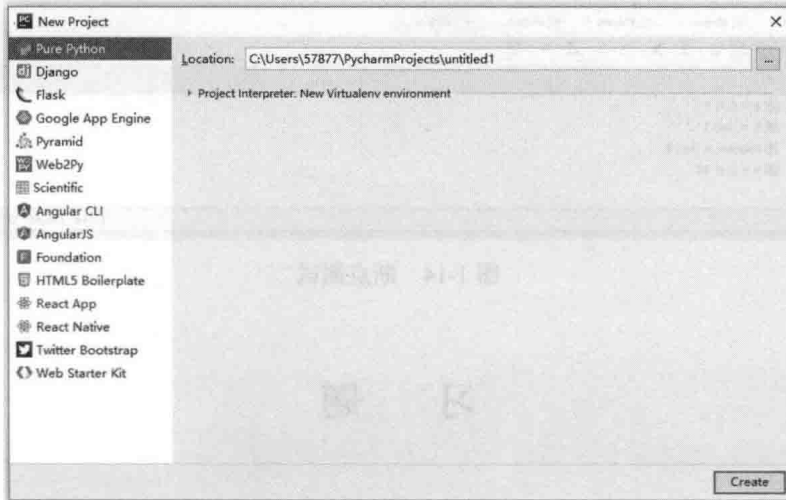


图 1-12 创建新文件

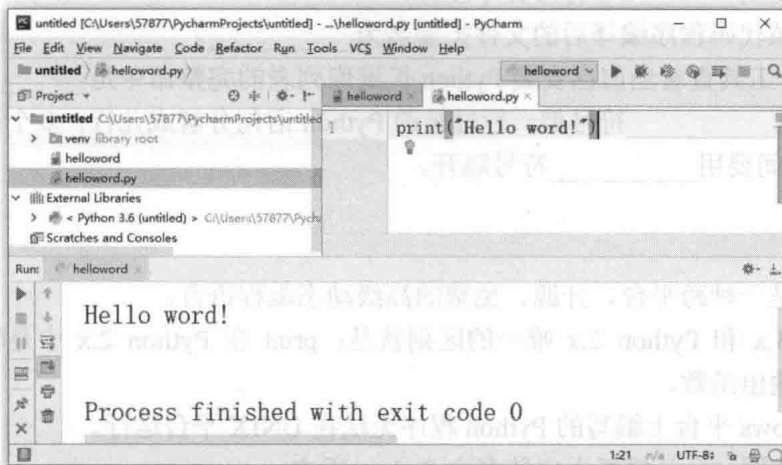


图 1-13 编辑、运行代码

单击行号和代码之间的位置生成一个有红色标识的断点位置，然后运行 Run 菜单项下的 Debug 命令就可以进行断点调试。在界面下方会显示该断点之前的变量信息。单击窗口中间部分的三角箭头按钮，可以进行逐步调试，如图 1-14 所示。

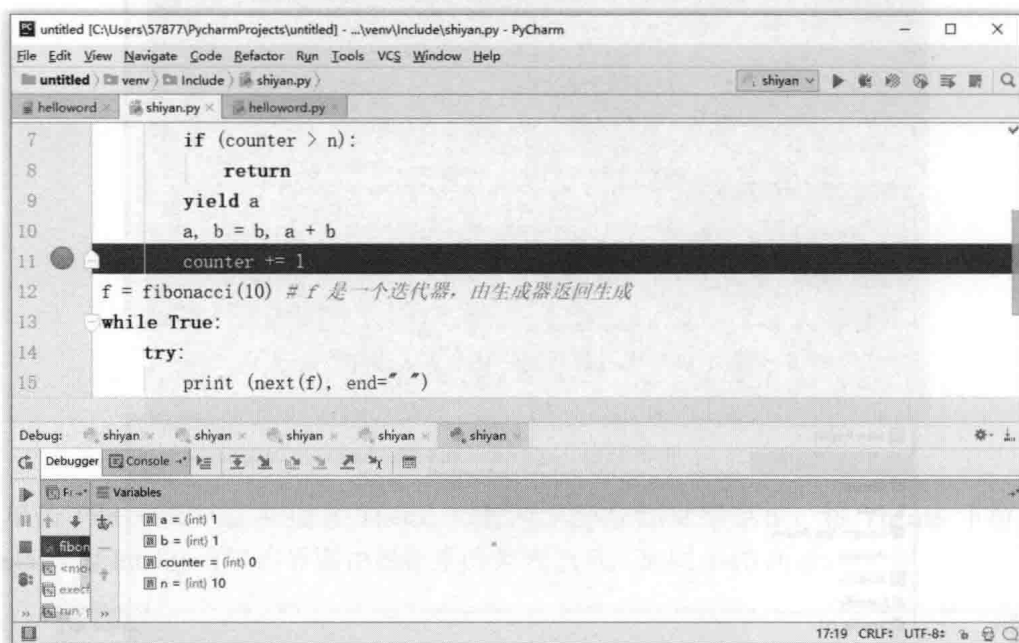


图 1-14 断点测试

习 题

一、填空题

1. Python 使用_____符号标示注释。
2. Python 源代码程序编译后的文件扩展名为_____。
3. 使用 pip 工具查看当前已安装 Python 扩展库列表的完整命令是_____。
4. 可以使用_____符号把一行过长的 Python 语句分解成几行；多个语句也可以写在同一行，语句之间要用_____符号隔开。

二、判断题

1. Python 是一种跨平台、开源、免费的高级动态编程语言。 ()
2. Python 3.x 和 Python 2.x 唯一的区别就是：print 在 Python 2.x 中是输出语句，而在 Python 3.x 中是输出函数。 ()
3. 在 Windows 平台上编写的 Python 程序无法在 UNIX 平台运行。 ()
4. 不可以在同一台计算机上安装多个 Python 版本。 ()
5. 在 Python 3.x 中可以使用中文作为变量名。 ()