COMPUTER

高等院校计算机
技术与应用系列规划教材

Python
程序设计教程

主　编　胡建华
副主编　李英杰　鄢　旭　宋广佳
刘福泉　王宇熙

ZHEJIANG UNIVERSITY PRESS
浙江大学出版社

高等院校计算机技术与应用系列规划教材

Python 程序设计教程

主　编　胡建华

副主编　李英杰　鄢　旭　宋广佳

刘福泉　王宇熙

ZHEJIANG UNIVERSITY PRESS
浙江大学出版社

图书在版编目（CIP）数据

Python 程序设计教程 / 胡建华主编. —杭州：浙江大学出版社，2019.3

ISBN 978-7-308-18989-7

Ⅰ.①P… Ⅱ.①胡… Ⅲ.①软件工具—程序设计—教材 Ⅳ.①TP311.561

中国版本图书馆 CIP 数据核字（2019）第 035612 号

Python 程序设计教程

主　编　胡建华

副主编　李英杰　鄢　旭　宋广佳　刘福泉　王宇熙

责任编辑　王元新

责任校对　徐　霞　李　晓

封面设计　刘依群

出版发行　浙江大学出版社

（杭州市天目山路 148 号　邮政编码 310007）

（网址：http://www.zjupress.com）

排　　版　杭州中大图文设计有限公司

印　　刷　绍兴市越生彩印有限公司

开　　本　787mm×1092mm　1/16

印　　张　9.5

字　　数　243 千

版 印 次　2019 年 3 月第 1 版　2019 年 3 月第 1 次印刷

书　　号　ISBN 978-7-308-18989-7

定　　价　29.00 元

前　言

本书主要作为高校计算机公共基础课——Python 语言程序设计的教材。随着人工智能的飞速发展，Python 语言以其简单实用、支持大量科学计算及智能算法库等特点迅速流行起来，2018 年的全国计算机等级考试也有了 Python 模块。目前，市面上 Python 的书籍已有很多，但大多数书籍内容较深，也有部分书籍内容组织不够严谨，语法体系不够清晰，不适合作为计算机公共基础课教材。

本书的特色是简洁明了，适用于文、理科各专业学生。本书紧扣全国计算机等级考试（二级）——Python 模块的主要内容，深入浅出地介绍 Python 语言的主要语法及编程的基本思想和方法；同时较详细地介绍 Python 计算生态常用库的安装及使用；最后通过单独一章综合实训，培养学生使用 Python 解决实际问题及进行数据分析的实战能力。本书的重点或难点章节均有视频教学资源，读者可以通过扫描二维码下载资源。每章后均有习题，以帮助读者更好地掌握各章知识点。本书既适合作为高校计算机公共基础课教材，也适合作为 Python 爱好者的入门教材。

本书共分为八章，包括 Python 语言概述，数据类型、运算符与表达式，程序控制结构，Python 序列与字典，Python 函数与模块，Python 文件操作，Python 常用模块，综合实训。

本书的编写分工为：胡建华编写第 1、2 章，刘福泉编写第 3 章，李英杰编写第 4、5 章，宋广佳编写第 6、7、8 章。全书由鄢旭、王宇熙审阅修订。

在本书编写过程中，参考了很多相关资料及网络文献，已在参考文献中列出。限于编者水平，书中难免存在不当之处，敬请广大读者指正。

编者

2018 年 11 月

目　录

第 1 章

Python 语言概述

本章主要讲述计算机的组成及其基本工作原理等基础知识、程序的概念、Python 的历史及特点、Python 的安装及基本使用方法；重点要理解程序的概念，掌握 Python 程序的基本结构及 IDLE 编辑器的使用。

1.1 计算机基础知识

1.1.1 计算机的发展历史

1946 年，世界上公认的第一台电子计算机 ENIAC(见图 1-1)诞生于美国的宾夕法尼亚大学。它使用的主要电子器件是电子管。它的诞生标志着现代电子计算机时代的来临。

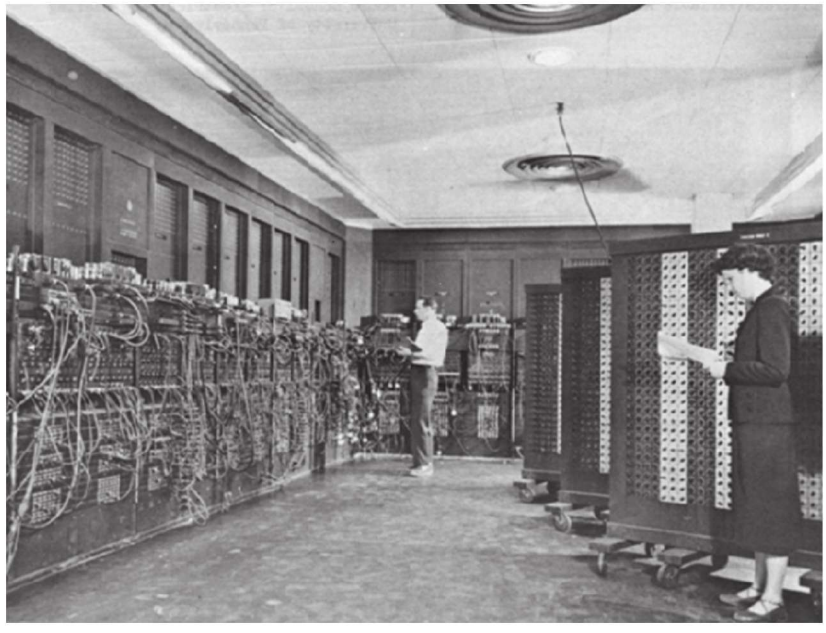

图 1-1 第一台电子计算机

按照采用的电子器件的不同，计算机分为 4 代：

第 1 代计算机(1946—1958 年)，其主要的电子器件是电子管。

第 2 代计算机(1959—1964 年)，其主要的电子器件是晶体管。

第 3 代计算机(1965—1970 年)，其主要的电子器件是中小规模集成电路。

第 4 代计算机(1971 年至今)，其主要的电子器件是大规模和超大规模集成电路。

目前，计算机的应用主要包括以下几个方面：

(1)科学计算(数值计算)。

(2)数据处理(信息管理)。

(3)过程控制(实时控制)。

(4)计算机辅助工程，主要包括计算机辅助设计(CAD)、计算机辅助制造(CAM)、计算机辅助教学(CAI)和计算机辅助测试(CAT)。

1.1.2 计算机系统的组成

美籍匈牙利科学家冯·诺依曼提出了计算机五大部件和存储程序思想。五大部件指运算器、控制器、存储器、输入设备和输出设备。存储程序思想指把计算机的工作过程描述为由许多命令按照一定的顺序组成的程序，然后把程序和数据一起输入计算机，计算机对已存入的程序和数据处理后，输出结果。

一个完整的计算机系统包括硬件系统和软件系统两大部分，如图 1-2 所示。

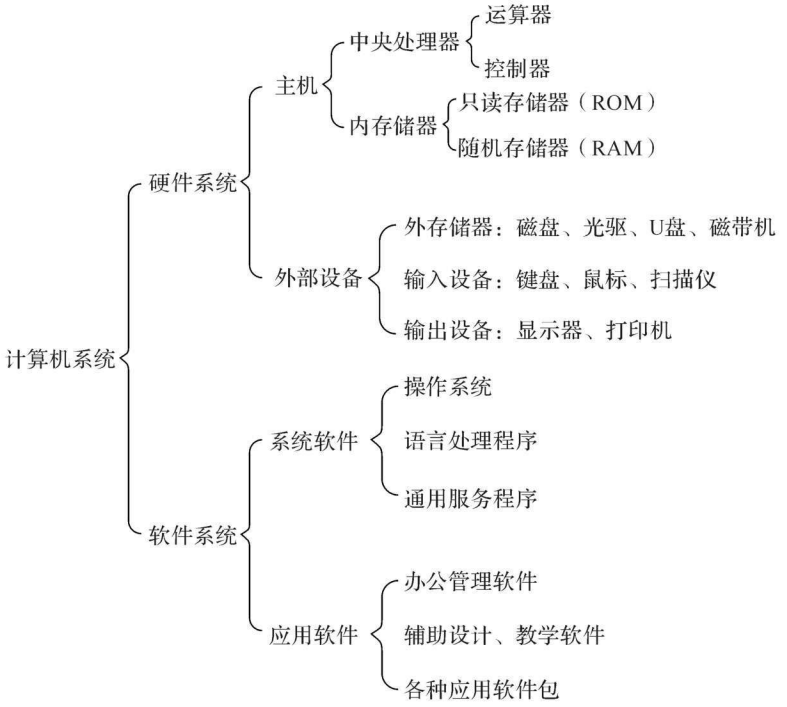

图 1-2 计算机系统组成

1. 硬件系统

硬件是组成一台计算机的各种物理装置。硬件系统包括运算器、控制器、存储器、输入设备和输出设备五大部分。通常，把运算器和控制器合在一起称为中央处理器，中央处理器和主存储器合在一起称为主机，输入设备和输出设备合称为外部设备。

2. 软件系统

软件是计算机运行所需要的各种程序、数据以及相关文档的总称。软件系统由系统软件和应用软件组成。

1.2　什么是程序

简单地说，程序就是指令的有序集合，是人们为了让计算机完成一个任务给计算机下达的命令集。例如，老师叫李明同学把教室的门关上，那么李明同学会怎么做呢？他为了完成这个任务，要做如下动作：①站起来；②转向门；③走过去；④伸手关门；⑤转向座位；⑥走回去坐下。每个动作就是一个指令，把它们按照①－②－③－④－⑤－⑥的顺序排列起来，就能完成关门的任务，这就是程序。大家思考一下，如果不按照上面的顺序能完成关门任务吗？肯定是不行的，所以，指令的顺序是非常重要的，这体现了程序设计的逻辑性。在后面的章节，我们会学到顺序、分支、循环三种逻辑结构。当你遇到一个任务，能够利用编程语言的指令，通过上述三种逻辑结构完成该任务，那么你就学会了程序设计。

1.3　Python 的发展历史及特点

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，是由 Guido van Rossum 在 1989 年年底发明的，第一个公开发行版本发行于 1991 年。Python 源代码遵循 GPL(General Public License，通用公共许可证)协议。

由于历史原因，Python 目前存在 Python 2.x 与 Python 3.x 两个版本。Python 3.0 版本常被称为 Python 3000，简称 Py3k，相对于 Python 的早期版本，这是一个较大的升级。为了不带入过多的累赘，Python 3.0 在设计时没有考虑向下兼容。许多针对早期 Python 版本设计的程序都无法在 Python 3.0 上正常执行。为了照顾现有程序，Python 2.6 作为一个过渡版本，基本使用了 Python 2.x 的语法和库，同时考虑了向 Python 3.0 的迁移，允许使用部分 Python 3.0 的语法与函数。由于 Python 3.x 版本功能设计更合理，所以目前主流应用都采用 Python 3.x 系列，全国计算机等级考试(二级)Python 模块也采用 Python 3.x 系列。本书采用了 Python 3.5 版本。

Python 语言具有以下特点：

(1)易于学习：Python 有相对较少的关键字，结构简单，学习起来十分轻松。

(2)易于阅读：Python 代码定义十分清晰。

(3)易于维护：Python 的源代码相当容易维护。

(4)一个广泛的标准库：Python 的优势之一是具有丰富的库，并且是跨平台的，在 Unix、Windows 和 Mac OS X 兼容很好。

(5)互动模式：您可以从终端输入执行代码并获得结果，互动地测试和调试代码片断。

(6)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。

(7)可扩展:如果你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,你可以使用 C 或 C ++ 完成那部分程序,然后从你的 Python 程序中调用。

(8)数据库:Python 提供所有主要的商业数据库的接口。

(9)GUI 编程:Python 支持 GUI 编程,可以移植到多个系统中。

(10)可嵌入:可以将 Python 嵌入到 C 或 C ++ 程序,让用户获得“脚本化”的能力。

1.4 Python 的安装

1-1 演示 Python 的下载及安装

Python 3. x 可安装在多个平台上,包括 Windows、Linux 和 Mac OS X 等。本书使用 Windows 平台。

1.4.1 下载 Python 安装程序

打开 Web 浏览器访问 https://www. python. org/downloads/windows,下载 executable installer,x86 表示 32 位 OS,x86-64 表示 64 位 OS,如图 1-3 所示。

图 1-3 下载界面

1.4.2 设置环境变量

右键单击“计算机”,然后单击“属性”,再点击“高级系统设置”;双击“系统变量”窗口下面的“Path”;然后在“Path”行添加 Python 安装路径(比如 D:\Python 32)。注意:路径直接用分号“;”隔开,如图 1-4 所示。

图 1-4　设置环境变量

设置成功以后，在 cmd 命令行输入命令“Python”，就可以执行 Python。

如表 1-1 所示为几个重要的环境变量。

表 1-1　重要的环境变量

变量名	描述
PYTHONPATH	PYTHONPATH 是 Python 搜索路径，默认输入的模块都会从 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 PYTHONCASEOK 的环境变量，就会使 Python 导入模块时不区分大小写
PYTHONHOME	另一种模块搜索路径。它通常内嵌于 PYTHONSTARTUP 或 PYTHONPATH 目录中，使得两个模块库更容易切换

1.5　IDLE 使用

IDLE 是 Python 软件包自带的一个集成开发环境，利用它可以方便地创建、运行、测试和调试 Python 程序。

1.5.1　IDLE 的安装

实际上，IDLE 是跟 Python 一起安装的，不过安装时要确保选中了“Tcl/Tk”组件，准确

地说，应该是不要取消选中，因为默认该组件是处于选中状态的。

1.5.2 IDLE 的启动

安装 Python 后，可以从“开始”菜单→“所有程序”→“Python 3.5”→“IDLE(Python GUI)”来启动 IDLE。IDLE 启动后的初始窗口如图 1-5 所示。

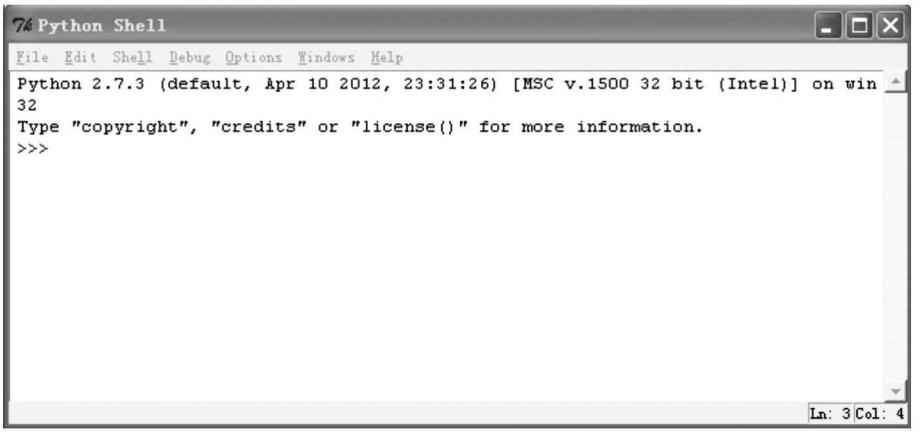

图 1-5 IDEL 界面

启动 IDLE 后，首先映入眼帘的是它的 Python Shell，我们通过它可以在 IDLE 内部执行 Python 命令。除此之外，IDLE 还带有一个编辑器用来编辑 Python 程序(或者脚本)，一个交互式解释器用来解释执行的 Python 语句，一个调试器用来调试 Python 脚本。

1.5.3 利用 IDLE 编辑器创建 Python 程序

IDLE 为开发人员提供了许多有用的特性，如自动缩进、语法高亮显示、单词自动完成以及命令历史等，在这些功能的帮助下，能够有效地提高开发效率。要新建一个文件，首先从“File”菜单中选择“New Window”菜单项，这样就可以在出现的窗口中输入程序的代码了。现在就让我们输入下面的代码来亲自体验一下 IDLE 编辑器所提供的各种便利吧。

示例程序的源代码如下：

```
#提示用户进行输入
x = input('请输入一个整数:')
x = int(x)
y = input('请再次输入一个整数:')
y = int(y)
if x > y:
    print('%d > %d' %(x,y))
else:
    print ('%d <= %d' %(x,y))
```

(1)自动缩进。实际上,很少有哪种语言能像 Python 这样重视缩进了,在其他语言比如 C 语言,缩进对于代码的编写来说是“有了更好”,而不是“没有不行”,它充其量是一个个人书写代码的风格问题;但是到了 Python 语言,则把缩进提升到了一种语法的高度。复合语句不是用大括号“{}”之类的符号表示,而是通过缩进来表示。这样做的好处就是减少了程序员的自由度,有利于统一风格,使得人们在阅读代码时会更加轻松。为此,IDLE 提供了自动缩进功能,它能将光标定位到下一行的指定空距处。当我们键入与控制结构对应的关键字,如 if 等时,按下回车键后 IDLE 就会启动自动缩进功能,如图 1-6 所示。

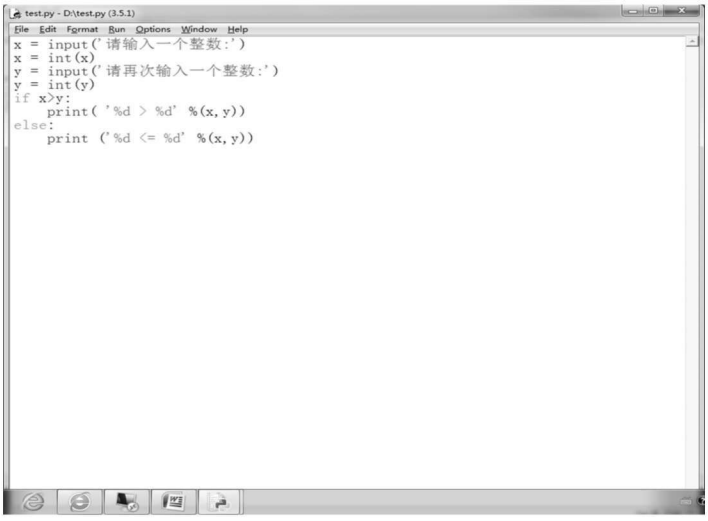

图 1-6　IDEL 编辑界面

当我们在 if 关键字所在行的冒号后面按回车键之后,IDLE 自动进行了缩进。一般情况下,IDLE 将代码缩进一级,即 4 个空格。如果想改变这个默认的缩进量的话,可以从“Format”菜单选择“New indent width”项来进行修改。对初学者来说,需要注意的是,尽管自动缩进功能非常方便,但是我们不能完全依赖它,因为有时候自动缩进未必完全符合我们的心意,所以还需要仔细检查。

(2)语法高亮显示。语法高亮显示就是给代码中不同的元素使用不同的颜色显示。默认时,关键字显示为橘红色,注释显示为红色,字符串显示为绿色,定义和解释器输出显示为蓝色,控制台输出显示为棕色。在键入代码时,会自动应用这些颜色突出显示。语法高亮显示的好处是,可以更容易区分不同的语法元素,从而提高可读性;与此同时,语法高亮显示还降低了出错的可能性。比如,如果输入的变量名显示为橘红色,那么您就需要注意了,这说明该名称与预留的关键字有冲突,所以必须给变量更换名称。

(3)单词自动完成。单词自动完成是指当用户输入单词的一部分后,从“Edit”菜单选择“Expand word”菜单项,或者直接按 Alt+/组合键自动完成该单词。有时候我们只记住了函数的开头几个字母,这时该怎么办?没关系,从“Edit”菜单选择“Show completetions”菜单项,IDLE 就会给出一些提示。这时只要按下回车键,IDLE 就会自动完成此函数名。如果不合适的话,还可以通过“↑”“↓”方向键进行查找。

创建好程序之后,从“File”菜单中选择“Save”保存程序。如果是新文件,会弹出“Save

as”对话框，我们可以在该对话框中指定文件名和保存位置。保存后，文件名会自动显示在屏幕顶部的蓝色标题栏中。如果文件中存在尚未存盘的内容，标题栏的文件名前后会有星号(*)出现。

1.5.4 常用编辑功能详解

对于“Edit”菜单，除了上面介绍的几个选项之外，其他常用的选项及解释如下：

Undo：撤销上一次的修改。

Redo：重复上一次的修改。

Cut：将所选文本剪切至剪贴板。

Copy：将所选文本复制到剪贴板。

Paste：将剪贴板的文本粘贴到光标所在位置。

Find：在窗口中查找单词或模式。

Replace：替换单词或模式。

Go to line：将光标定位到指定行首。

对于“Format”菜单，常用的选项及解释如下：

Indent region：使所选内容右移一级，即增加缩进量。

Dedent region：使所选内容左移一级，即减少缩进量。

Comment out region：将所选内容变成注释。

Uncomment region：去除所选内容每行前面的注释符。

New indent width：重新设定制表位缩进宽度，范围为 2～16，宽度为 2 相当于 1 个空格。

Expand word：单词自动完成。

Toggle tabs：打开或关闭制表位。

1.5.5 在 IDLE 中运行 Python 程序

要使用 IDLE 执行程序的话，可以从“Run”菜单中选择“Run Module”菜单项，该菜单项的功能是执行当前文件。对于前面的示例程序，执行情况如图 1-7 所示。

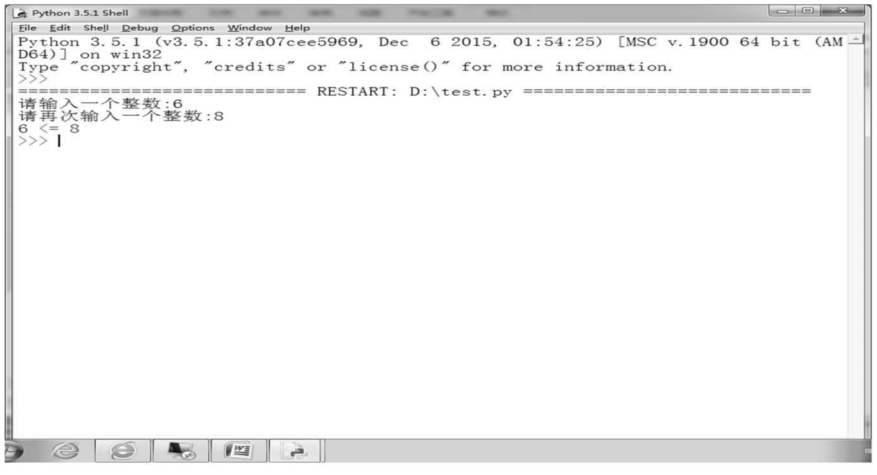

图 1-7 IDLE 运行界面