

Python

零基础快乐学习之旅

洪锦魁◎编著

(K12实战训练)

200个精彩程序实例

银行复利计算

计算地球到月球所需时间

摄氏度与华氏度转换

体重与健康判断程序

用户账号管理系统

购物车设计

总分平均名次成绩系统设计

真心认识Tuple

设计英汉与汉英字典

文件搜索与分析

夏令营的程序设计

威力彩与大乐透程序

认识赌场的游戏骗局

程序调试典故

尾牙兑奖程序

清华大学出版社

Python

零基础快乐学习之旅

(K12实战训练)



本书特色

这是一本为完全没有Python基础的人所写的编程书籍。由于篇幅所限，又要完整表达Python的奥妙与精华，笔者为内容的取舍与程序范例的选择投入了相当多的精力。笔者凭借多年的著作经验，深知**程序逻辑思维**从实际工程项目开始，为此笔者精心地设计了许多通俗易懂的程序实例，期待读者在笔者的引导下，以最短的时间学会Python，为进入**人工智能、机器学习、大数据**等领域做好准备。

图书查询·扩展阅读



书圈

清华社官方微信号



扫我有惊喜

ISBN 978-7-302-53254-5



9 787302 532545 >

定价：59.00元

Python

零基础快乐学习之旅 (K12实战训练)

洪锦魁◎著



清华大学出版社
北京

内 容 简 介

本书在讲解 Python 编程语言语法概念的同时融入了相关的科学知识。随着人工智能技术的飞速发展,编程教育越来越重要。编程的核心是算法和逻辑,是通往未来的语言。近期,国务院发布《新一代看人工智能发展规划》,大力推广 K12 编程教育,还有的省已经将信息技术纳入高考科目。本书内容涵盖 Python 的专题设计和案例,是 K12 实战训练的指导教程。

Python 是目前较热门也是功能较强大的程序语言。本书除了对 Python 语言基本程序语法内容解说,还融入了程序设计的逻辑思维,希望读者可以完全吸收,未来可以活用这个功能强大的程序语言。

本书各章末都辅以专题设计,这些精彩、实用的专题程序实例,可以让读者充分体会各种语法的定义与精神,同时增强程序设计的逻辑思维能力。

为了方便学校教师教学,本书所有习题均有习题解答。

本书适合高中生(含)或初学 Python 语言编程者阅读。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Python 零基础快乐学习之旅: K12 实战训练 / 洪锦魁编著. —北京: 清华大学出版社, 2019
ISBN 978-7-302-53254-5

I. ①P… II. ①洪… 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 134527 号

责任编辑: 杨迪娜 栾大成

封面设计: 杨玉兰

责任校对: 徐俊伟

责任印制: 宋 林

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 三河市龙大印装有限公司

经 销: 全国新华书店

开 本: 170mm×240mm 印 张: 15.75 字 数: 342 千字

版 次: 2019 年 10 月第 1 版 印 次: 2019 年 10 月第 1 次印刷

定 价: 59.00 元

产品编号: 082950-01

序

多次与教育界的朋友沟通，谈到计算机语言的发展趋势，大家一致认为 Python 已经是当今非常重要的计算机程序设计语言。目前，Microsoft、Facebook、Google 等知名公司都已经将此语言列为员工必会的计算机语言。

世界各国已纷纷将 Python 列为学生阶段必学的程序语言之一，为了让学生或是计算机初学者也加入到学习 Python 的行列，笔者尝试将 Python 语法的各种用法用简单并辅以丰富活泼、精彩、实用的程序实例的方式解说。为了让读者更精通 Python 的用法，每章最后皆辅以专题设计习题，这些设计可以充分培养读者程序设计的逻辑思维能力。

本书用 200 个程序实例讲解了下列知识：

- 变量与基本数学运算 - 专题：银行存款复利的计算
- Python 的基本数据类型 - 专题：计算地球到月球所需时间
- 基本输入与输出 - 专题：摄氏度与华氏度的转换
- 程序流程控制 - 专题：人体体重与健康判断程序
- 列表 - 专题：用户账号管理系统
- 循环设计 - 专题：创建真实的成绩系统
- 元组 - 专题：认识元组
- 字典 - 专题：遍历字典
- 集合 - 专题：夏令营的程序设计
- 函数设计 - 专题：用函数重新设计记录一篇文章每个单词的出现次数
- 面向对象 - 专题：解说函数与方法
- 设计与应用模块 - 专题：认识赌场游戏骗局
- 文档的读取与写入 - 专题：文档搜索

- 程序调试与异常处理 - 专题：认识程序调试的典故
- 排序与搜寻 - 专题：尾牙兑奖号码设计

洪锦魁编写过多部计算机图书。本书保持笔者一贯的特色，程序实例丰富。相信读者只要遵循本书内容的学习，就可以在最短时间，用最快乐的方式学会 Python 程序设计，编著本书虽力求完美，但难免学习经历不足，出现谬误，请读者不吝指正。

目 录

第 1 章 基本概念

1-1 认识 Python	2
1-2 Python 的起源	2
1-3 Python 语言发展史	3
1-4 Python 的应用范围	4
1-5 跨平台的程序语言	4
1-6 系统的安装与执行	4
1-7 文档的创建、存储、执行与打开	5
1-7-1 文档的创建	5
1-7-2 文档的存储	6
1-7-3 文档的执行	6
1-7-4 打开文档	7

第 2 章 认识变量与基本数学运算

2-1 用 Python 做计算	10
2-2 认识变量	10
2-3 认识程序的意义	12
2-4 认识注释的意义	13
2-4-1 注释符号 #	13
2-4-2 三个单引号或双引号	13
2-5 Python 变量与其他程序语言的差异	14
2-6 变量的命名原则	14
2-7 基本数学运算	15
2-7-1 四则运算	15
2-7-2 余数和整除	16
2-7-3 乘方	16
2-7-4 Python 语言控制运算的优先级	16
2-8 指派运算符	17
2-9 Python 等符号的多重指定使用	18
2-10 Python 的断行	18
2-11 专题设计：银行存款复利的计算	19

第 3 章 Python 的基本数据类型

3-1 type() 函数	23
3-2 数值数据类型	24
3-2-1 整数与浮点数的运算	24
3-2-2 强制数据类型的转换	25
3-2-3 数值运算常用的函数	25
3-3 布尔值数据类型	27
3-4 字符串数据类型	27
3-4-1 字符串的连接	28
3-4-2 处理多于一行的字符串	29
3-4-3 逸出字符	29
3-4-4 强制转换为字符串 str()	30
3-4-5 字符数据的转换	30
3-4-6 聪明地使用字符串加法和换行字符 \n	31
3-5 专题设计：计算地球到月球所需时间	31

第 4 章 基本输入与输出

4-1 Python 的辅助说明 help()	36
4-2 格式化输出数据使用 print()	36
4-2-1 函数 print() 的基本语法	36
4-2-2 格式化 print() 输出	37
4-2-3 精准控制格式化的输出	38
4-2-4 format() 函数	39
4-3 数据输入 input()	40
4-4 专题设计：摄氏度和华氏度的转换	41

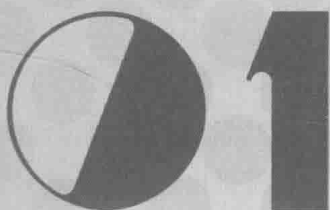
第 5 章 程序的流程控制使用 if 语句

5-1 关系运算符	46
5-2 逻辑运算符	47
5-3 if 语句	48

5-4	if ... else 语句.....	50	6-8	再谈字符串.....	80
5-5	if ... elif ...else 语句.....	52	6-8-1	字符串的索引.....	80
5-6	巢状的 if 语句.....	53	6-8-2	字符串切片.....	81
5-7	专题设计：人体体重与健康判断 程序.....	54	6-8-3	函数或方法.....	82
第6章 列表 (list)			6-8-4	将字符串转成列表.....	82
6-1	认识列表 (list).....	60	6-8-5	使用 split() 处理字符串.....	82
6-1-1	列表基本定义.....	60	6-8-6	字符串的其他方法.....	83
6-1-2	读取列表元素.....	61	6-9	in 和 not in 语句.....	85
6-1-3	列表切片 (list slices).....	63	6-10	专题设计：用户账号管理系统... ..	87
6-1-4	列表索引值是 -1.....	64	第7章 循环设计		
6-1-5	列表统计资料、最大值 max()、最 小值 min()、总和 sum().....	64	7-1	基本 for 循环.....	94
6-1-6	列表个数 len().....	65	7-1-1	for 循环基本操作.....	95
6-1-7	更改列表元素的内容.....	66	7-1-2	如果程序代码段只有一行.....	96
6-1-8	列表的相加.....	66	7-1-3	有多行的程序代码段.....	96
6-1-9	删除列表元素.....	67	7-1-4	将 for 循环应用在数据 类型的判断.....	97
6-1-10	列表为空列表的判断.....	67	7-2	range() 函数.....	98
6-2	Python 简单的面向对象概念.....	68	7-2-1	只有一个参数的 range() 函数.....	99
6-2-1	字符串的方法.....	68	7-2-2	扩充专题银行存款复利的轨迹... ..	99
6-2-2	更改字符串大小写.....	70	7-2-3	有两个参数的 range() 函数.....	100
6-3	增加与删除列表元素.....	70	7-2-4	有 3 个参数的 range() 函数.....	101
6-3-1	在列表末端增加元素 append()	70	7-2-5	一般应用.....	102
6-3-2	插入列表元素 insert().....	71	7-2-6	设计删除列表内所有元素.....	103
6-3-3	删除列表元素 pop().....	71	7-3	进阶的 for 循环应用.....	103
6-3-4	删除指定的元素 remove().....	72	7-3-1	巢状 for 循环.....	103
6-4	列表的排序.....	73	7-3-2	强制离开 for 循环 - break 指令... ..	105
6-4-1	颠倒排序 reverse().....	73	7-3-3	for 循环暂时停止不往下 执行 - continue 指令.....	106
6-4-2	sort() 排序.....	73	7-4	while 循环.....	107
6-5	进阶列表操作.....	75	7-4-1	基本 while 循环.....	108
6-5-1	index().....	75	7-4-2	巢状 while 循环.....	109
6-5-2	count().....	75	7-4-3	强制离开 while 循环 - break 指令.....	109
6-6	列表内含列表.....	76	7-4-4	while 循环暂时停止不往下 执行 - continue 指令.....	110
6-6-1	基本概念.....	76	7-4-5	while 循环条件语句与可迭代 对象.....	111
6-6-2	再看二维列表.....	77	7-5	专题设计：购物车设计.....	111
6-7	列表的赋值与复制.....	79			
6-7-1	列表赋值.....	79			
6-7-2	列表的复制.....	80			

7-6 专题设计：创建真实的成绩系统.....	112
第 8 章 元组 (tuple)	
8-1 元组的定义.....	120
8-2 读取元组元素.....	121
8-3 遍历所有元组元素.....	121
8-4 修改元组内容产生错误的实例...	122
8-5 可以使用全新定义方式修改元组元素.....	122
8-6 元组切片 (tuple slices).....	123
8-7 方法与函数.....	123
8-8 列表与元组数据互换.....	124
8-9 其他常用的元组方法.....	125
8-10 元组的功能.....	126
8-11 专题设计：认识元组.....	126
第 9 章 字典 (dict)	
9-1 字典基本操作.....	131
9-1-1 定义字典.....	131
9-1-2 列出字典元素的值.....	132
9-1-3 增加字典元素.....	133
9-1-4 更改字典元素内容.....	133
9-1-5 删除字典特定元素.....	134
9-1-6 删除字典所有元素.....	134
9-1-7 删除字典.....	134
9-1-8 创建一个空字典.....	135
9-1-9 字典的复制.....	135
9-1-10 取得字典元素数量.....	136
9-1-11 验证元素是否存在.....	136
9-1-12 设计字典的可读性技巧.....	137
9-2 遍历字典.....	138
9-2-1 遍历字典的键 - 值.....	138
9-2-2 遍历字典的键.....	139
9-2-3 依键排序与遍历字典.....	141
9-2-4 遍历字典的值.....	141
9-2-5 依值排序与遍历字典的值.....	142
9-3 字典内键的值是列表.....	143
9-4 while 循环在字典的应用.....	144
9-5 字典常用的函数和方法.....	145
9-5-1 len().....	145
9-5-2 get().....	145
9-6 专题设计：记录一篇文章每个单词的出现次数.....	146
第 10 章 集合 (set)	
10-1 创建集合.....	152
10-1-1 使用大括号创建集合.....	152
10-1-2 使用 set() 函数定义集合.....	153
10-1-3 大数据与集合的应用.....	155
10-2 集合的操作.....	155
10-2-1 交集 (intersection).....	156
10-2-2 并集 (union).....	157
10-2-3 差集 (difference).....	158
10-2-4 关键词 in.....	159
10-3 专题设计：夏令营的程序设计 ...	160
第 11 章 函数设计	
11-1 Python 函数基本概念.....	164
11-1-1 函数的定义.....	165
11-1-2 没有输入参数也没有返回值的函数.....	165
11-2 函数的参数设计.....	166
11-2-1 传递一个参数.....	167
11-2-2 多个参数传递.....	167
11-2-3 参数默认值的处理.....	168
11-3 函数返回值.....	168
11-3-1 返回 None.....	169
11-3-2 简单返回数值数据.....	170
11-3-3 返回多种数据的应用.....	170
11-3-4 简单返回字符串数据.....	171
11-4 调用函数时参数是列表.....	171
11-5 传递任意数量的参数.....	172
11-5-1 传递并处理任意数量的参数....	172
11-5-2 设计含有一般参数与任意数量参数的函数.....	173
11-6 局部变量与全局变量.....	174
11-6-1 全局变量可以在所有函数使用.....	174
11-6-2 局部变量与全局变量使用相同的名称.....	174

11-6-3 程序设计需要注意事项.....	175	14-1-2 读取整个文档 read().....	204
11-7 匿名函数 lambda.....	176	14-1-3 with 关键词.....	205
11-8 专题设计：用函数重新设计记录 一篇文章每个单词的出现次数 ...	177	14-1-4 逐行读取文档内容	206
第 12 章 类别一面向对象		14-1-5 逐行读取使用 readlines()	207
12-1 类别的定义	182	14-2 写入文档.....	208
12-2 类别的属性与方法.....	183	14-2-1 将执行结果写入空的文档内 ...	208
12-3 专题设计：解说函数与方法	183	14-2-2 输出多行数据的实例	209
第 13 章 设计与应用模块		14-2-3 创建附加文档.....	210
13-1 将自建的函数存储在模块中	186	14-3 专题设计：文档搜索.....	210
13-1-1 事前准备工作.....	186	第 15 章 程序调试与异常处理	
13-1-2 创建函数内容的模块	187	15-1 程序异常.....	216
13-2 应用自己创建的函数模块	187	15-1-1 一个除数为 0 的错误	216
13-2-1 import 模块名称	188	15-1-2 撰写异常处理程序 try - except.....	216
13-2-2 导入模块内特定单一函数.....	188	15-1-3 try - except - else.....	219
13-2-3 导入模块内多个函数	189	15-1-4 找不到文档的错误 FileNotFoundError.....	219
13-2-4 导入模块内所有函数	189	15-2 常见的异常对象	220
13-3 随机数 random 模块.....	190	15-3 finally	222
13-3-1 randint().....	190	15-4 专题设计：认识程序调试的典故... ..	223
13-3-2 choice()	191	第 16 章 算法 - 排序与搜寻	
13-3-3 shuffle().....	192	16-1 算法 (algorithm).....	227
13-3-4 sample().....	193	16-2 排序 (sort).....	228
13-4 时间 time 模块	193	16-3 搜寻 (search).....	230
13-4-1 time().....	193	16-3-1 顺序搜寻法 (sequential search).....	230
13-4-2 sleep().....	194	16-3-2 二分搜寻法 (binary search).....	231
13-4-3 asctime().....	195	16-4 专题设计：尾牙兑奖号码设计 ...	233
13-4-4 localtime().....	195	附录 A 安装 Python	236
13-5 日期 calendar 模块	196	A-1 Windows 操作系统的 Python 安装	237
13-5-1 列出某年是否是闰年 isleap()... ..	196	附录 B ASCII 码值表	240
13-5-2 输出月历 month()	196		
13-5-3 输出年历 calendar().....	197		
13-6 专题设计：认识赌场游戏骗局... ..	197		
第 14 章 文档的读取与写入			
14-1 读取文档.....	203		
14-1-1 打开文档 open() 与关闭 文档 close()	203		



第 1 章

基本概念

本章摘要

- 1-1 认识 Python
- 1-2 Python 的起源
- 1-3 Python 语言发展史
- 1-4 Python 的应用范围
- 1-5 跨平台的程序语言
- 1-6 系统的安装与执行
- 1-7 文档的创建、存储、执行与打开

1-1 认识 Python

Python 是一种直译式 (interpreted language)、面向对象 (object oriented language) 的程序语言, 它拥有完整的函数库, 可以协助开发人员轻松地完成许多常见工作。

所谓的直译式语言是指, 直译器 (interpreter) 会将程序代码一句一句的直接执行, 不需要经过编译 (compile) 动作, 将语言先转换成机器码, 再予以执行。目前它的直译器是 CPython, 这是由 C 语言编写的一个直译程序, 与 Python 一样目前是由 Python 基金会管理使用。

Python 也是一个动态的高级语言, 具有垃圾回收 (garbage collection) 功能, 所谓的垃圾回收是指程序执行时, 直译程序会主动收回不再需要的动态内存空间, 将内存集中管理, 这种机制可以减轻程序设计师的负担, 当然也就减少了程序设计师犯错的机会。

由于 Python 是一个开放的原始码 (open source), 每个人皆可免费使用或为它贡献资源, 除了它本身有许多内建的套件 (package) 或模块 (module), 许多公司也为它开发了更多的套件, 促使它的功能可以持续扩充, 这也是本书的主题。根据 IEEE Spectrum 在 2018 年发布的程序语言排名, Python 继续保持 2017 年时的排名, 位列第一。

1-2 Python 的起源

Python 的最初设计者是吉多·范罗姆苏 (Guido van Rossum), 他是荷兰人, 1956 年出生于荷兰哈勒姆, 1982 年毕业于阿姆斯特丹大学的数学和计算器系, 获得硕士学位。

1996 年吉多·范罗姆苏在为 Mark Lutz 所著的 *Programming Python* 的序言中表示: “1989 年我想在圣诞节期间思考设计一种程序语言打发时间, 当时正在构思一个新的脚本 (Script) 语言的解释器, 它是 ABC 语言的后代, 期待这个程序语言对 UNIX C 的程序语言设计师会有吸引力。基于我是蒙提派森飞行马戏团 (Monty Python's Flying Circus) 的疯狂爱好者, 所以就以 Python 为名称, 当作这个程序的名字。”

很多 Python 的文档中或某些图书封面喜欢用蟒蛇代表 Python, 从吉多·范罗姆苏的上述序言可知, Python 灵感的来源是马戏团名称而非蟒蛇。

1999 年他向美国国防高级研究计划局 DARPA (Defense Advanced Research Projects Agency) 提出 Computer Programming for Everybody 的研发经费申请, 他提出

了下列 Python 的目标。

- 这是一个简单直觉式的程序语言，可以和主要程序语言一样强大。
- 这是开放源代码（open source），每个人皆可自由使用与贡献。
- 程序代码像英语一样容易理解与使用。
- 可在短期间内开发一些常用功能。

现在上述目标皆已经实现了，Python 已经与 C/C++、Java 一样成为程序设计师必会的程序语言，然而它比 C/C++ 和 Java 更容易学习。

有关新版软件下载相关信息可以在 Python 软件基金会获取，可参考附录 A。

1-3 Python 语言发展史

1991 年 Python 正式诞生，当时的操作系统平台是 Mac。尽管吉多·范罗姆苏承认 Python 的设计是基于 ABC 语言，但是 ABC 语言并没有成功，吉多·范罗姆苏本人认为 ABC 语言并不是一个开放的程序语言，是失败的主要原因。因此，在 Python 的推广中，他避开了这个错误，将 Python 推向开源系统，而获得了巨大的成功。

□ Python 2.0 发表

2000 年 10 月 16 日 Python 2.0 正式发表，主要是增加了垃圾回收的功能，同时支持 Unicode 编码规则。

Unicode 是一种适合多语系的编码规则，主要功能是使用可变长度字节方式存储字符，以节省内存空间。例如，对于英文字母而言，使用 1 个字节空间存储即可，对于含有附加符号的希腊文、拉丁文或阿拉伯文等则用 2 个字节空间存储字符，中文字符则是以 3 个字节空间存储字符，只有极少数的平面辅助文字需要 4 个字节空间存储字符。也就是说，这种编码规则已经包含了全球语言的字符了，当采用这种编码方式设计程序时，其他语系的程序只要支持 Unicode 编码规则皆可显示。例如：法国人即使用法文版的程序，也可以正常显示出中文字符。

□ Python 3.0 发表

2008 年 12 月 3 日，Python 3.0 正式发表。一般程序语言的发展会考虑到兼容特性，但是 Python 3.0 在开发时为了不受到先前 2.x 版本的束缚，没有考虑兼容特性，所以许多早期版本开发的程序是无法在 Python 3.x 版上执行的。

为了解决这个问题，尽管发表了 Python 3.0 版本，在后来将 3.0 版本的特性移植到 Python 2.6/2.7x 版本上。

Python 基金会提醒：Python 2.7x 已经被确定为最后一个 Python 2.x 的版本。

本书是以 Python 3.x 版本为撰写依据。

1-4 Python 的应用范围

Python 是一个非常适合初学者学习的程序语言，在国外有许多儿童程序语言教学也是以 Python 为工具，它同时还是一个功能强大的程序语言，下列是它的部分应用。

- 设计动画游戏。
- 支持图形用户接口 (Graphical User Interface, GUI) 开发。
- 数据库设计、开发与管理网站。
- 执行科学运算与大数据分析。
- Google、Yahoo!、YouTube、NASA、Dropbox (文档分享服务)、Reddit (社交网站) 在内部皆大量使用 Python 做开发工具。
- 网络爬虫、黑客攻防。

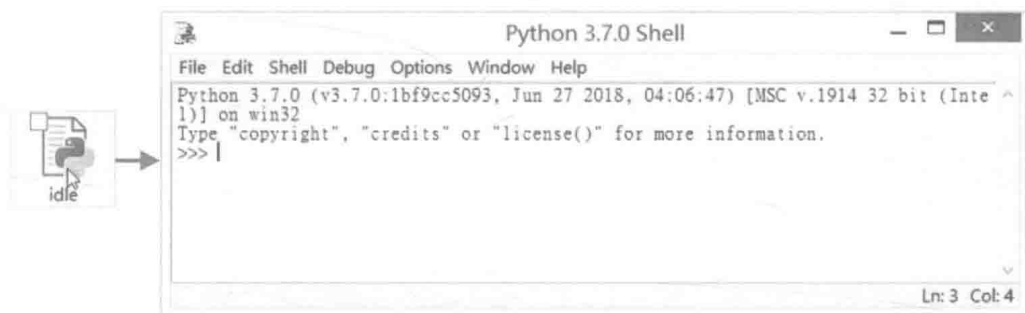
1-5 跨平台的程序语言

Python 是一种跨平台的程序语言，主要操作系统 (Windows、Mac OS、UNIX/Linux 等) 皆可以安装和使用。

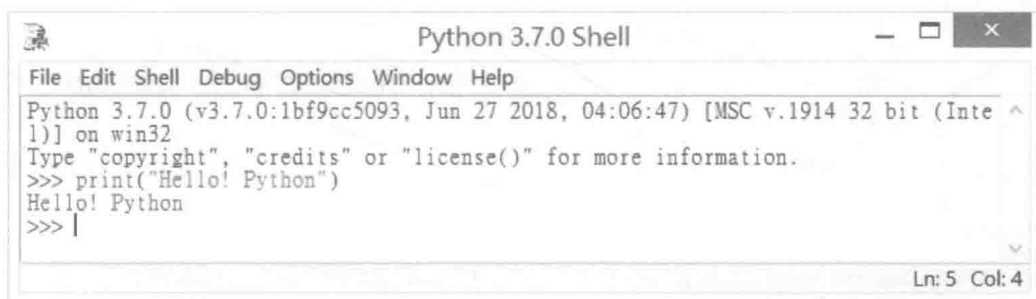
跨平台的程序语言意味着用户可以在某一个平台上使用 Python 设计一个程序，未来这个程序也可以在其他平台上顺利使用。

1-6 系统的安装与执行

相关安装 Python 的步骤请参考附录 A。下面将以 Python 3.7x 版本为例做说明。请双击在附录 A 所建，在 Windows 桌面上的 idle 图标，将看到 Python 3.7.0 Shell 窗口。



上述 `>>>` 符号是提示信息，可以在此输入 Python 指令，下图是一个简单的 `print()` 函数，目的是输出字符串，单引号或双引号间的文字称为字符串。



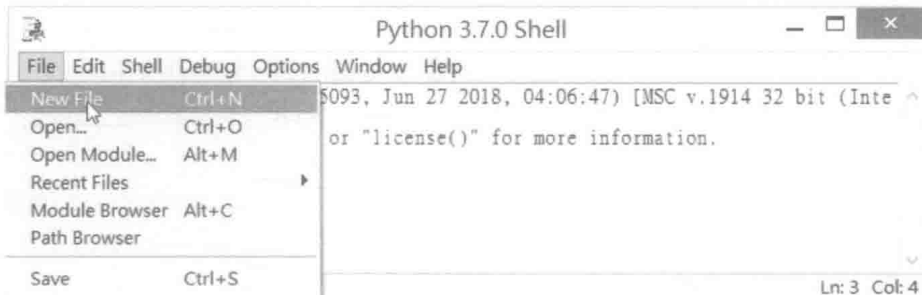
由上图可以确定我们成功地执行第一个 Python 的程序实例了。

1-7 文档的创建、存储、执行与打开

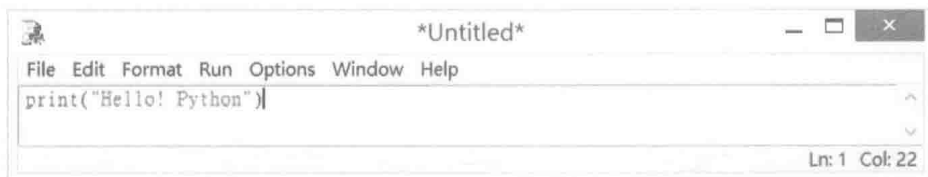
如果设计一个程序每次都要在 Python Shell 窗口环境重新输入指令的话，是一件很麻烦的事，当程序设计时，将所设计的程序保存在文档内是一件重要的事。

1-7-1 文档的创建

在 Python Shell 窗口可以执行 `File/New File`，创建一个空白的 Python 文档。



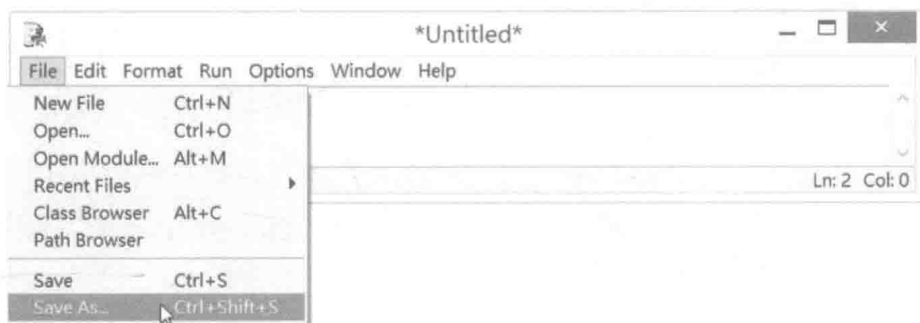
然后创建一个 Untitled 窗口，窗口内容是空白，如下所示是笔者在空白文档内输入一道指令的实例。



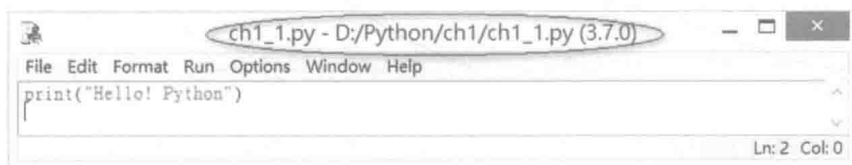
如果想要执行上述文档，需要先存储上述文档。

1-7-2 文档的存储

可以执行 File/Save As 存储文档。



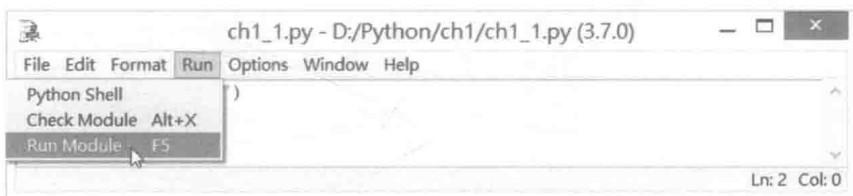
然后将看到另存新文档的对话框，此例笔者将文档存储在 D:/Python/ch1 文件夹，文件名是 ch1_1 (Python 的扩展名是 py)，请在文件名中输入 ch1_1，再按保存按钮，可以得到下列结果。



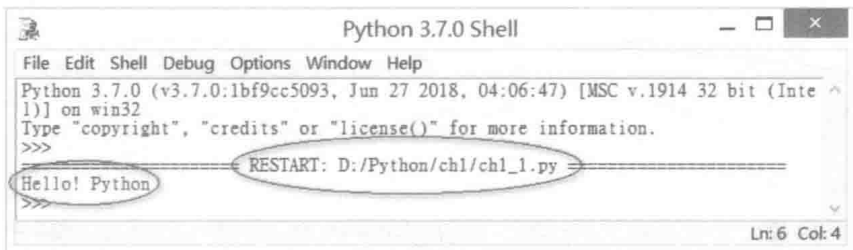
可以看到原标题 Untitled 已经改为 ch1_1.py 了。

1-7-3 文档的执行

执行 Run/Run Module，就可以正式执行先前所建的 ch1_1.py 文档。



执行后，在原先的 Python Shell 窗口可以看到执行结果。



学习到此，恭喜你已经成功地创建一个 Python 文档，同时执行成功了。

1-7-4 打开文档

假设已经离开 ch1_1.py 文档，未来想要打开这个程序文档，可以执行 File/Open。



然后会出现打开旧文档对话框，请选择欲打开的文档即可。

习题

一、是非题

1. (×). 使用 Python 需要付费买授权。(1-1 节)
2. (×). Python 在执行前需要先编译，将程序转成可执行文档然后才可以执行。(1-1 节)
3. (O). Python 是面向对象 (object oriented) 的程序语言。(1-1 节)
4. (×). Python 在 3.0 版开始支持垃圾回收和 Unicode 编码规则。(1-3 节)
5. (O). 可以使用 Python 设计动画游戏、动态网页设计、网络爬虫。(1-4 节)
6. (O). Python 是一种跨平台语言。(1-5 节)

二、选择题

- 1 (A). Python 的发明与哪一个人有关? (1-2 节)