# Python 编程从入门到实践

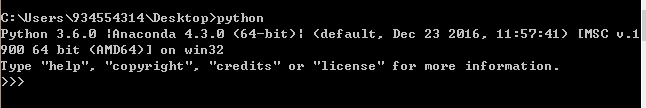
## 起步

### 1.1搭建编程环境

(1)linux系统自带python环境，

### 1.2 在不同的操作系统中搭建Python 编程环境

（1）检查Python版本，直接输入python命令就可以。



（2）安装文本编辑器，常用的文本编辑器有 subline vscode ,pycharm 等

（3）在window环境中安装python强烈建议安装anaconda，自带Python的各种工具库

### 1.3 解决安装问题

（1）程序出现错误的时候，python将显示trackback，会提供一些线索，让你知道是什么问题让程序无法运行

（2）在网上寻找答案

### 1.4 在终端运行python程序

（1）python hello\_world.py直接在终端运行程序

（2）window获取文件列表为dir,linux获取文件列表为ls

## 第二章 变量和简单数据类型

### 2.1 运行hello\_word.py时发生的情况

（1）编辑器将使用Python解释器读取整个程序，确定每个单词的含义，编写程序时，编辑器会以各种方式突出程序的不同部分，这种功能叫语法突出。

### 2.2变量

（1）每个变量都存储了一个值，与变量相关联的信息，在程序中可能随时修改变量的值，python将始终记录变量的最新值。

（2）变量名只能包含字母、数字和下划线，变量名可以字母或者下划线开头，不能以数字开头。1\_message，这种是不允许的。

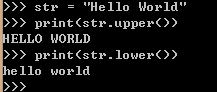
（3）不要将python关键字和函数名作为变量名。

（4）变量名应该既简短又有描述性，student\_name

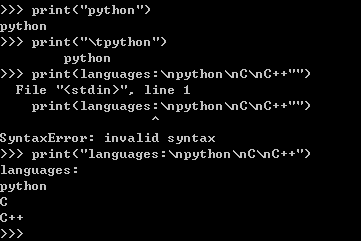
（5）慎用小写字母i和大写字母O，容易看错。

### 2.3字符串

（1）字符串就是一系列的字符，在python中可以使用单引号，也可以使用双引号，如‘this is a string’或者“this is a string”。

（2）str.upper()将字符串改为全部大写，str.lower()将字符串改为全部小写 

（3）字符串的合并 str = str1 + str2

（4）使用制表符或者换行来添加空白，

（5）删除多余的空白



### 2.4 数字

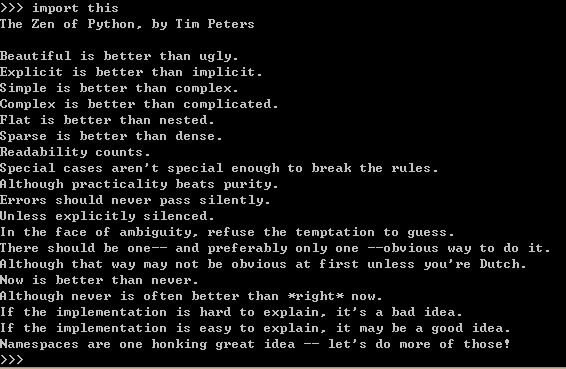
（1）python中执行加减乘除，平方等

### 2.5注释

（1）写注释的主要目的是阐述代码是要做什么，以及如何做的。

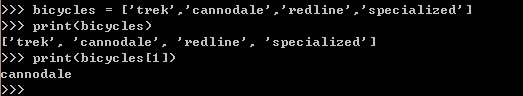
### 2.6 python之禅

（1）编程是要解决问题的，设计良好、高效而漂亮的解决方案都会让程序员心生敬意。



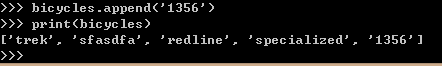
## 第三章 列表简介

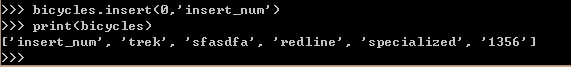
### 3.1列表是什么

（1）由一些列特定顺序的元素组成，列表的索引是从0开始，而不是从1开始。

### 3.2 修改、添加和删除元素

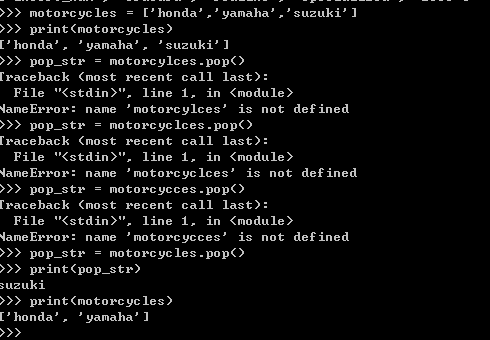
（1）直接对某一个数进行赋值，就可以修改列表。

（2）使用append, 

（3）使用Insert对列表进行插入，

（4）del删除列表，

（5）使用pop删除列表中的元素，并接着使用它的值，



（6）根据值来删除元素，直接用remove

### 3.3组织列表

（1）使用sort()方法对列表进行永久性的排序

（2使用函数sorted()对列表进行临时排序

（3）顺序列表取反，reverse()方法

（4）获取列表的长度len(list)

### 3.4 使用列表时避免索引错误

（1）索引都是从0开始，获取最后一个，直接用List[-1]

## 第四章 操作列表

### 4.1遍历整个列表

（1）使用for循环打印列表中的所有数据，for list in lists:

### 4.2避免缩进错误

（1）python根据缩进来判断代码行与前一个代码行的关系，

### 4.3创建数值列表

（1）for value in range(1,5)

Print(value)

### 4.4使用列表的一部分

（1） 处理部分元素，俗称切片，list[0:3]

### 4.5元组

（1）不可变的列表成为元组，dimensions = (200,10),使用的是圆括号。

### 4.6设置代码格式

（1）缩进每级缩进都使用4个空格，提高可读性。

（2）行长不超过80字符。

（3）将程序不同部分分开，使用空行。

## 第五章 If语句

### 5.1一个简单的示例

（1）if a == b :

else:

### 5.2 条件测试

（1）if a == b and c > = d

### 5.3 if语句

### 5.4 使用if语句处理列表

## 第六章 字典

### 6.1 一个简单的字典

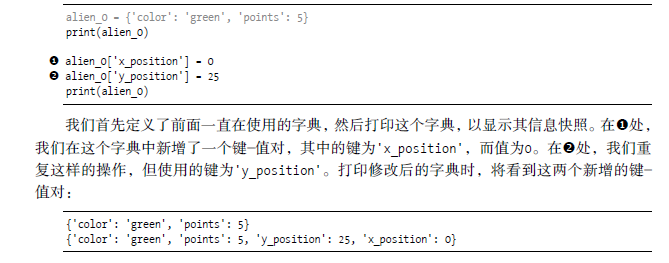
（1）字典是一些列键-值对。

alien\_0 ={‘color’:’green’,’points’:5}

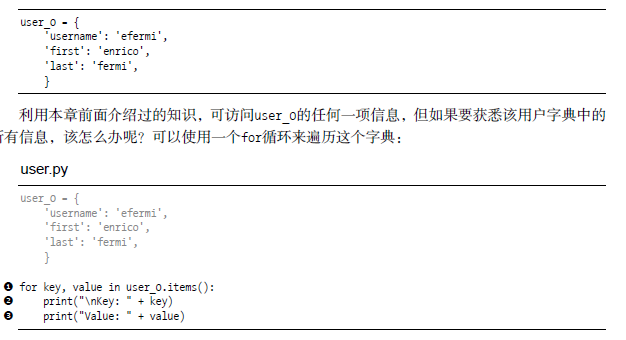
Print(alien\_0[‘color’])

Print(alien\_0[‘points’])

### 6.2 使用字典

（1）

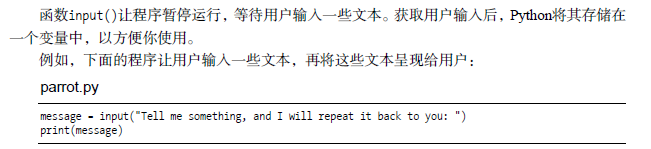
### 6.3 遍历字典

（1）

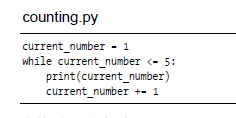
### 6.4 嵌套

## 第七章 用户输入和while循环

### 7.1 函数input()的工作原理

（1）

### 7.2 while循环简介

（1）

### 7.3 使用while循环来处理列表和字典

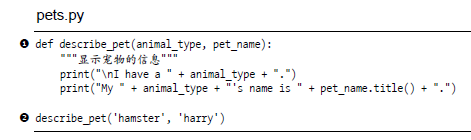
（1）

## 第八章 函数

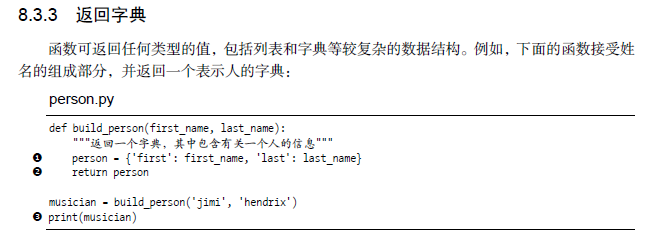
### 8.1 定义函数

（1）

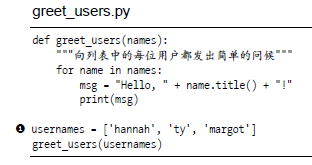
### 8.2 传递实参

（1）

### 8.3 返回值

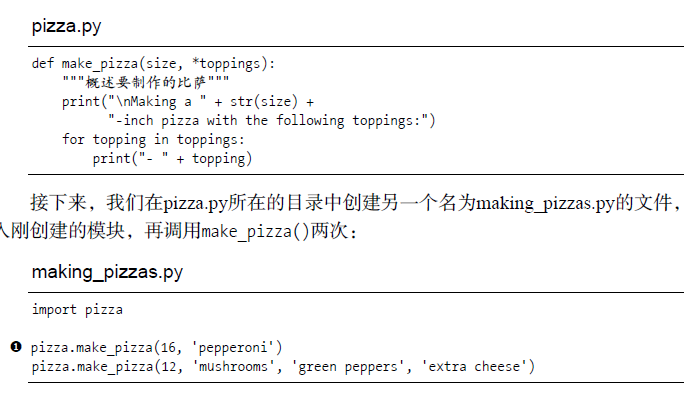
（1）

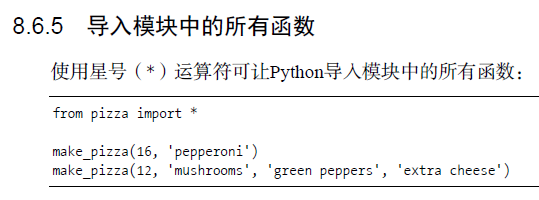
### 8.4 传递列表

（1）

### 8.5 传递任意数量的实参

### 8.6 将函数存储在模块中

（1）

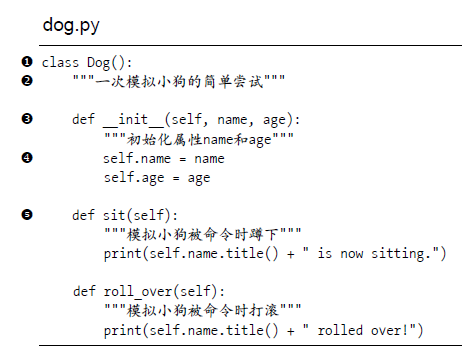
（2）

### 8.7 函数编写指南

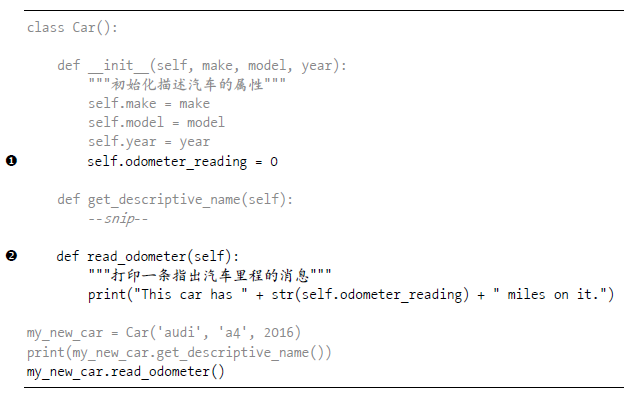
（1）给形参指定默认值时，等号两边不要有空格，

## 第九章 类

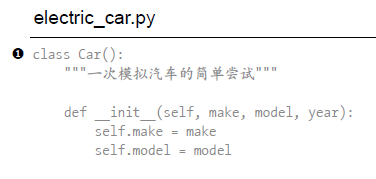
### 9.1 创建和使用类

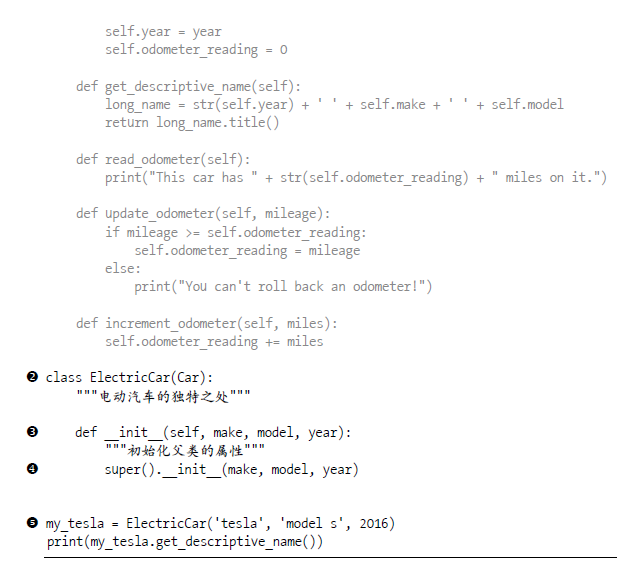
（1）

### 9.2 使用类和实例

（1）

### 9.3 继承

（1）



### 9.4 导入类

（1）from car import Car,ElectricCar

（2）创建整个模块，import car

### 9.5 python标准库

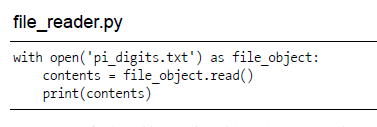
（1）import collections

### 9.6 类编码风格

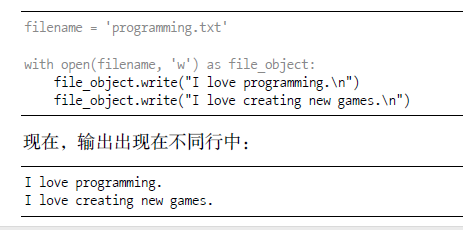
（1）类名中的每个单词的首字母都大写，而不使用下划线，实例名和模块名都采用小写格式，并在单词之间增加下划线。

## 第十章 文件和异常

### 10.1 从文件中读取数据

（1）

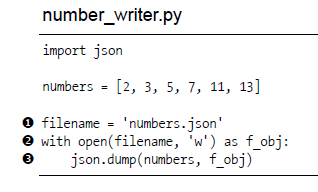
### 10.2 写入文件

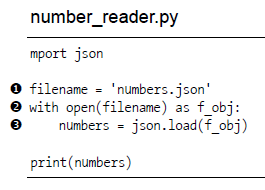
（1）

### 10.3 异常

（1）

### 10.4 存储数据

（1）存储数据

（2）加载数据

## 第十一章 测试代码

### 11.1测试函数

### 11.2 测试类

## 第十二章 武装飞船

### 12.1 规划项目

### 12.2 安装pygame

### 12.3 开始游戏项目

### 12.4 添加飞船图像

### 12.5 重构：模块game\_functions

### 12.6驾驶飞船

### 12.7 简单回顾

### 12.8 射击

## 第十三章 外星人

### 13.1 回顾项目

### 13.2 创建第一个外星人

### 13.3创建一群外星人

### 13.4让外星人群移动

### 13.5 射杀外星人

### 13.6结束游戏

### 13.7确定应运行游戏的那些部分

## 第十四章 记分

### 14.1添加Play按钮

### 14.2提高等级

### 14.3记分

## 第十五章 生成数据

### 15.1 安装matolotlib

### 15.2 绘制简单的折线图

### 15.3 随机漫步

### 15.4 使用pygal模拟投掷骰子

## 第十六章 下载数据

### 16.1 CSV文件格式

### 16.2制作世界人口地图：JSON格式

## 第十七章 使用API

### 17.1使用web API

### 17.2使用pygal可视化仓库

### 17.3 hacker news API

## 第十八章Django入门

### 18.1 建立项目

### 18.2 建立应用程序

### 18.3 创建网页：学习笔记主页

### 18.4 创建其他网页

## 第十九章 用户账户

### 19.1让用户能够输入数据

### 19.2创建用户账户

### 19.3 让用户拥有自己的数据

## 第二十章 设置应用程序的样式并对其部署

### 20.1 设置项目“学习笔记”的样式

### 20.2 部署“学习笔记”

## 附录A 安装python

## 附录B 文本编辑器

## 附录C 寻求帮助

## 附录D 使用git进行版本控制