# Python

# 第一个Python程序

## 输出与输入

### 输出指令print

print ‘’,’’,’’：指令后引号内既是输出的内容，可调用，可做算法，可直接填写；

>>>print 'abcd','1',2,3+4,50/2,50\*0.2,((5+3)-2)\*3

>>>abcd 1 2 7 25 10.0 18。

**注：**在python2中print是调用并非函数，而python3中归入了函数范畴，因此在python3中print的函数的书写格式就变为了print()的格式。

### 输入指令raw\_input()

name = raw\_input("shuru")：提示并进入待输入，输入内容将为name变量赋值；

### 输入指令raw\_input()与input()的区别

**raw\_input()：**输入任何内容都是字符串(String)格式；

**input()：**在python规则内，输入的是什么格式就会是什么格式；

**注：**输入指令，输入到了那里去，首先要创建变量，并给这个变量定义，可理解为创建一个空文档，你输入不同的后缀，赋予了他到底是什么格式。

### 输入输出相互结合

提示输入给name变量赋值，并打印预设内容及调用name内容。

>>>name = raw\_input("shuru")

>>>print "hello", name

# 二、Python基础

## （一）、转义符

### \：改变特定字符本来含义，则需在该字符前追加 “ \ ” 处理；

### 例如：print 'I\'m \"OK\"!'，取消了标色字符的本来含义，只是正常的打印了出来；

### \n：换行的转义符；

### \t：标识制表符；

### \\：转义自身；

### r’’：不转义符，引号内无论打什么都不转义；

### ’’’…’’’：换行简化，换行较多时可用’’’圈起内容，内容正常换行即可，无需\n；

## （二）、布尔值true/false

布尔值只有一个唯一的结果，不是True就是False，既“真”或“假”，如2>1既为True，反之若1>2则会返回False；

### 布尔值运算符：

### **and：**当含有and运算符时，所有判断均为True时才会返回True，否则为False，如1>2 and 2>1 则为False，1<2 and 2>1 则为True；

### **or：**当含有or运算符时，所有的判断中有1个为True时就会返回True，否则为False，如1>2 or 2>1 则为True，1>2 or 2>3 则为False；

### **not：**当运算前赋予not运算，则会将后续运算结果反之，如not 1>2，则为True；

### **None：**空值，既没有值，空值不等于“0”；

### **%：**取余运算符，如10 % 3，意为10÷3=3余1，得到最终结果为10 % 3=1取余数；

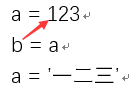
### 注：加减乘除中，除法需区分整数除还是浮点数除，如10/3，10为整数，除3自然结果也会得到整数，既10/3=3，而想做精确除法，则需将10变为浮点数，既10.0/3=3.3333333；

## （三）、变量

### 变量定义：变量既是一个可被赋值各种类型的容器，例如a=1，既为创建变量a，并创建一个整数1，将a指向到了1，完成对a的变量赋值，对变量的赋值可以赋值任意类型，如整数、浮点数、字符串、布尔值等；

### 等号含义：在python中“=”等号是赋值语句，可将任意数据类型赋值给变量，且1个变量可被反复赋值，且可为不用类型。

### 变量赋值变量：当1个变量赋值到另一个变量时，赋值到的不是变量本身，而是被赋值变量当时的值，如下所示，逐条执行后，b依旧等于123，只是a的赋值从整数123变为了字符串的一二三；、

****

### 常量：常量就是大写的变量，是约定俗成的，不可变的变量；

## （四）、函数

### len()：返回字符串的长度，若对数据长度进行判断验证，可将数据编码转为Unicode进行判断，单纯的使用len(‘中文’)=4个字符串，而len(u’中文’)=2个字符串；

### ord/chr：对基础字母、数字、简单符号进行ASCII编码的转换，如用ord(‘A’)，将字符串A转换后既为65，而用chr(65)转换后得到的就是字符串’A’；

## （五）、编码

### 文本与终端：运行终端是运行终端，而.python文本是python文本，在使用终端运行python文本时，使用的是ASCII编码，ASCII编码不直接支持中文，而python文本中表明“# -\*- coding: utf-8 -\*-”，是表明文本内使用的是utf-8编码，utf-8支持中文；

### 编码说明： ASCII编码可以看做是utf-8编码，但是Unicode只能是Unicode，不过可以在Unicode与utf-8之间进行转换；

### encode('utf-8')方法：将以Unicode编码为格式的中文字符串，转换成utf-8的编码格式，如“u'中文'.encode('utf-8')”，转换后所得“'\xe4\xb8\xad\xe6\x96\x87'”；

### decode('utf-8')方法：将以utf-8编码为格式的中文编码格式，转换成Unicode编码格式，如

|  |
| --- |
| >>> 'abc'.decode('utf-8')  u'abc'  >>> '\xe4\xb8\xad\xe6\x96\x87'.decode('utf-8')  u'\u4e2d\u6587'  >>> print '\xe4\xb8\xad\xe6\x96\x87'.decode('utf-8')  中文 |

### 注：只需确定两个条件，“你的脚本编码格式” 和 “输出终端的编码格式”，如果这两个编码格式不一致，就需要转换，并且转换都是先将数据decode成unicode编码，然后encode成你的终端编码格式后运行。

## （六）、格式化

### 输出格式化字符串：'亲爱的xxx你好！你xx月的话费是xx，余额是xx'，其中每一处的XX均是一个变量值，所以需要一种简便的格式化字符串，简化的方式就是使用“%”百分号运算符；

### %d：用整数替换；

### %f：用浮点数替换；

### %s：用字符串替换；

### %x：用十六进制整数替换；

### 例1：有几个%@占位符，就需要跟几个变量或具体值，若只有1个可省略括号，如下，有2个%s1个%d就需要2个字符串和1个整数；

### 'Hi, %s%s, you have $%d.' % ('Michael','Lucifer',1000000)

### 'Hi, MichaelLucifer, you have $1000000.'

### 例2：

### %2d-%02d' % (3, 1)：隔2个字符显示整形3,隔2个字符位并用0代替，结果为’3-01’；

### '%.2f' % 3.1415926：四舍五入提取至浮点后两位；

### u'Hi, %s' % u'Michael'：替换Unicode编码字符串，尽量所替字符串也是Unicode编码；

### 注：如果不确定使用什么，%s可作为通用的运算符，把任何数据类型转为字符串；

### 注：当%需要普通字符含义时，只需%%即可；

## （七）、使用list

### List：list（序列/列表/数组）是一种数据结构，可对数组的内容进行调整，也可往数组中添加或删除，或者检查是否存在于数组之中；

### 列表内容位置：列表内容是由0、1、2……的数字进行位置标记的；

### 列表内容类型：可以是字符串，可以是整形，可以是浮点型，可以是布尔值均可，可以是二维数组，也可是已赋值的数组；

>>> p = ['asp', 'php']

>>> s = ['python', 'java', p, 'scheme']

### 如何增减内容：

### 以下说明以 shuzu = [‘No1’,’NO2’,’NO3’,2017,2018,2019] 和 shuzu2 = [‘guo’,’xiang’]作为参考；

### 输出：可直接对数组进行完整输出，print shuzu ，也可输出数组中某个位置的内容，print shuzu[0]或print shuzu[0:5]，既显示完整数组、显示数组第1个数据、显示数组第1个到第4个数据；

### 更新/替换：修改已有数据时，可直接索引其在数组中的位置，然后对其进行赋值，例如shuzu[0] = ‘No@’，既[‘No@’,’NO2’,’NO3’,2017,2018,2019]；

### 尾部添加值：使用append()方法来实现对数组的添加，shuzu.append(2020)，其执行结果为[‘No1’,’NO2’,’NO3’,2017,2018,2019,2020]；

### 索引位置添加：使用insert(index,obj)方法针对位置进行添加，如insert(0,’No0’)，其执行结果为[‘No0’,‘No1’,’NO2’,’NO3’,2017,2018,2019]；

### 数组合并：使用extend()方法，可以将两个1维数组合并成为1个，例如shuzu.extend(shuzu2)，其执行结果为shuzu = [‘No1’,’NO2’,’NO3’,2017,2018,2019, ‘guo’,’xiang’]，B会并入A，A数组产生更新，B不会有任何改变；

### 删除数组元素：可用del语句删除数组内容或整个数组，例如del.shuzu[0]，其执行结果为[’NO2’,’NO3’,2017,2018,2019]，若不做索引del.shuzu会将整个数组删掉；

注：也可使用shuzu.pop[]，不填值既代表默认删除末尾元素；

### N是否存在的判断：可用in方法验证，例如2019 in shuzu，其结果为True；

### 组合列表：可用 “+” 号操作符，对1个以上的数组进行组合，此只是单纯的组合展示，并不会改变原有数组，与extend()方法不同；

### 重复单一数组：可用 “\*”号操作符，对1个数组执行，会将该数组复制合并展示，但并不会改元原有数组；

### 比较数组：使用cmp(list1,list2)函数比较两个数组，一致返回0，否则返回1；

### 返回列表最大与最小：可使用max() 和 min() 函数；

**(七)、使用tuple**

**tuple（元组）：**与list（数组）不同，tuple可被使用，但初始赋值后就不可变了，