**数据类型概述**

数字

基本数字类型

int：有符号整数

bool：布尔值

true：1

false：0

float：浮点数

complex：复数

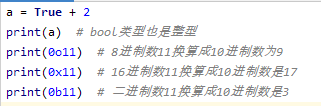
数字表示方式

python默认以十进制数显示

数字以0o或0O开头表示为8进制数

数字以0x或0X开头表示16进制数

数字以0b或0B开头表示为2进制数



字符串

定义字符串

python中字符串被定义为引号之间的字符集合

python支持使用成对的单引号或双引号

无论单引号还是双引号，表示的意义相同

python还支持三引号（三个连续的引号或者双引号），可以用来包含特殊字符

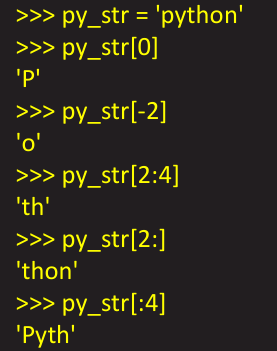
python不区分字符和字符串

字符串切片

使用索引运算符[]和切片运算符[:]可得到字符串

第一个字符串的索引是0，最后一个字符的索引是-1

子字符串包含切片中的起始下标，但不包含结合下标

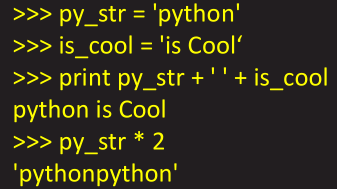


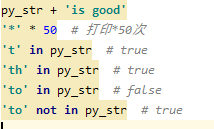


字符串连接操作

使用+号可以将多个字符串拼接在一起

使用\*号可以将一个字符串重复多次



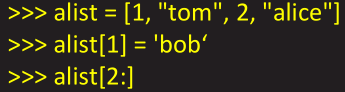


定义列表

可以将列表当成普通的“数组”，它能保存任意数量任意类型的python对象

像字符串一样，列表也支持下标喝切片操作

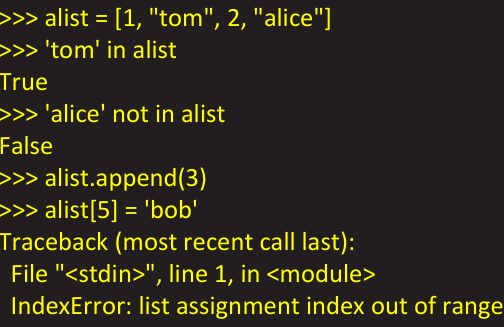
列表中的项目可以改变

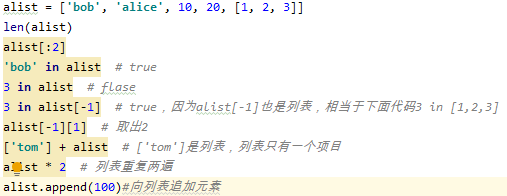


列表操作

使用in或not in判断成员关系

使用append方法向列表中追加元素

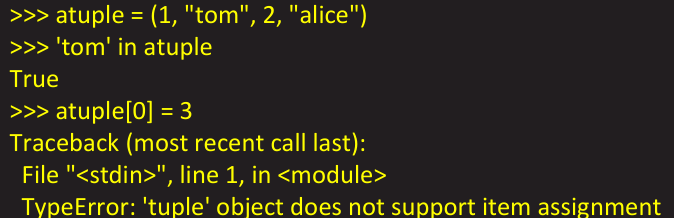


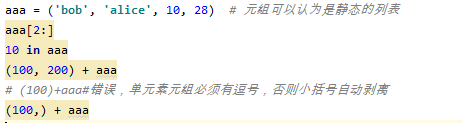


元组的定义及操作

可以认为元组是“静态”的列表

元组一旦定义，不能改变



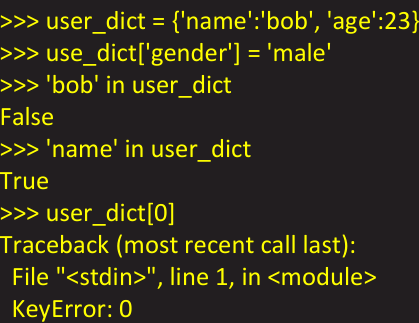


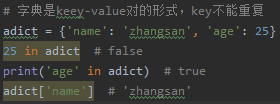
字典

字典的定义及操作

字典是由键-值对构成的映射数据类型

通过键取值，不支持下标操作





数据类型比较

按存储模型分类

标量类型：数值、字符串

容器类型：列表、元组、字典

按更新模型分类：

可变类型：列表、字典

不可变类型：数字、字符串、元组

按访问模型分类

直接访问：数字

顺序访问：字符串、列表、元组

映射访问：字典

**判断语句**

if语句

if语句语法结构

标准if条件语句的语法

if expression:

if\_suite

else:

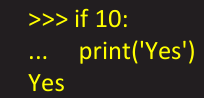
else\_suite

如果表达式的值非0或者为布尔值true，则代码组if suite被执行；否则就去执行else suite

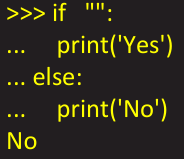
代码组是一个python术语，它由一条或多条语句组成，表示一个子代码块

if语句示例解析

只要表达式数字为非零值即为true



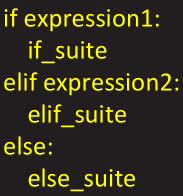
空字符串、空列表、空元组、空字典的值均为false



扩展if语句

扩展if语句结构

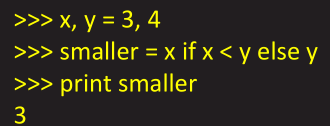
扩展if语句结构

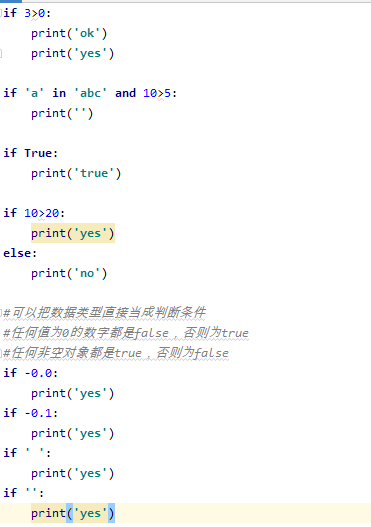


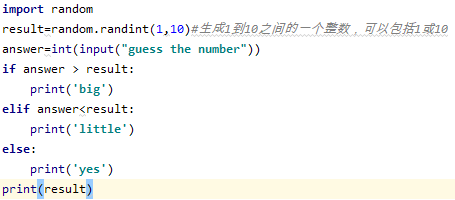
条件表达式

python在很长的一段时间里没有条件表达式（C?X:Y），或称三元运算符

从python2.5集成的语法确定为：X if C else Y









**while循环**

循环语句基础

循环概述

一组被重复执行的语句称之为循环体，能否继续重复，决定循环的终止条件

python中的循环有while循环和for循环

循环次数未知的情况下，建议采用while循环

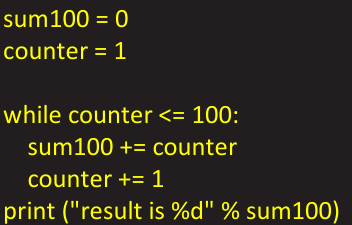
循环次数可以预知的情况下，建议采用for循环

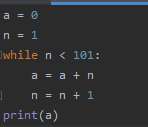
while循环语法结构

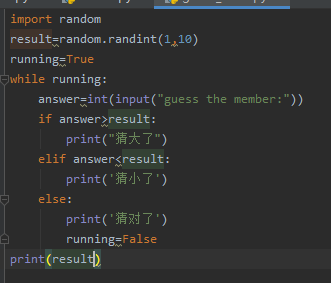
当需要语句不断的重复进行时，可以使用while循环



语句while\_suite会被连续不断的循环执行，直到表达式的值为-或FLASE





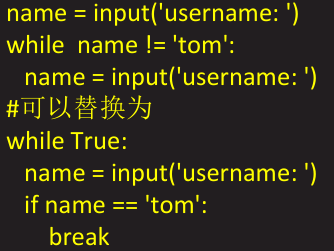


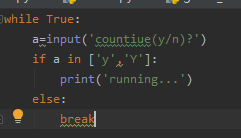
循环语句进阶

break语句

break语句可以技术当前循环然后跳转到下条语句

写程序的时候，应尽量避免重复的代码，在这种情况下可以使用while-break结构

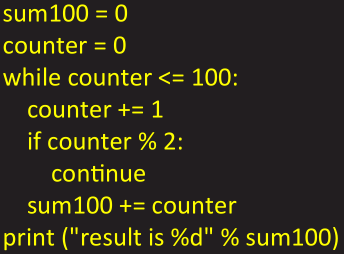


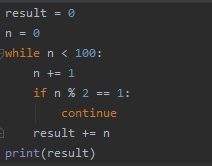


continue语句

当遇到contiye语句时，程序循环会终止当前循环，并忽略剩余的语句，然后回到循环的顶端

如果仍然满足循环条件，循环体内语句继续执行，否则退出循环





else语句

python中的while语句也支持else子句

else子句只在循环完成后执行

break语句也会跳过else块

