**Python格式化中如何使用%运算符**

本文只要是介绍python格式化在使用%运算符时，Python格式化如何使用%运算符，以下是相关内容的详细介绍。希望你会有所收获。

在python中你也许会遇到过python格式化的相关实际应用的内容，而在python格式化中你是否对其的实际应用步骤或是对其的代码有不解之处，如果存在相关的问题，你就可以浏览下面得文章。

在python中也有类似于c中的printf()的格式输出标记。在python格式化使用的是%运算符，通用的形式为

格式标记字符串 % 要输出的值组 其中，左边部分的”格式标记字符串“可以完全和c中的一致。右边的'值组'如果有两个及以上的值则需要用小括号括起来，中间用短号隔开。重点来看左边的部分。左边部分的最简单形式为：%cdoe 其中的code有多种，不过由于在python格式化中，所有东西都可以转换成string类型，因此，如果没有什么特殊需求完全可以全部使用’%s‘来标记。比如：

'%s %s %s' % (1, 2.3, ['one', 'two', 'three'])

它的输出为

'1 2.3 ['one', 'two', 'three']'

就是按照%左边的标记输出的。虽然第一个和第二值不是string类型，一样没有问题。在这个过程中，当电脑发现第一个值不是%s时，会先调用整型数的函数，把第一个值也就是1转成string类型，然后再调用str()函数来输出。前面说过还有一个repr()函数，如果要用这个函数，可以用%r来标记。除了%s外，还有很多类似的code:

整型数：%d 无符号整型数：%u 八进制：%o 十六进制：%x %X 浮点数：%f科学记数法: %e %E 根据数值的不同自动选择%e或%f: %g 根据数值的不同自动选择%E或%f: %G 就跟前面说用\进行转义一样，这里用%作为格式标记的标识，也有一个%本身应该如何输出的问题。如果要在”格式标记字符串“中输出%本身，可以用%%来表示。上面说的只是格式标记的最简间的形式，来看复杂一点的：

‘%6.2f’ % 1.235

在这种形式中，在f的前面出现了一个类似小数的6.2它表示的意思是，总共输出的长度为6个字符，其中小数2位。还有更复杂的：

‘%06.2f’ % 1.235

在6的前面多了一个0,表示如果输出的位数不足6位就用0补足6位。这一行的输出为‘001.24’，可以看到小数也占用一位。类似于这里0这样的标记还有-、+。其中，-表示左对齐，+表示在正数前面也标上+号，默认是不加的。最后来看最复杂的形式：

‘%(name)s:%(score)06.1f’ %{'score':9.5, 'name':'newsim'}

这种形式只用在要输出的内容为dictionary（一种python格式化的数据类型）时，小括号中的(name)和(score)对应于后面的键值对中的键。前面的例子可以看到，”格式标记字符串“中标记的顺序和"要输出的值组"中的值是一一对应的，有顺序，一对一，二对二。而在这种形式中，则不是，每个格式标记对应哪个值由小括号中的键来指定。这行代码的输出为：'newsim:0009.5'。

有时候在%6.2f这种形式中，6和2也不能事先指定，会在程序运行过程中再产生，那怎么输入呢，当然不能用%%d.%df或%d.%d%f。可以用%\*.\*f的形式，当然在后面的”要输出的值组“中包含那两个\*的值。比如：'%\*.\*f' % (6, 2, 2.345)就相当于'%6.2f' % 2.345。