**Python3.4中filter函数，map函数和reduce函数**

**三个函数比较类似，都是应用于序列的内置函数。常见的序列包括list、tuple、str。**

**1.map函数**

map函数会根据提供的函数对指定序列做映射。

map函数的定义：

map(function, sequence[, sequence, ...]) -> list

通过定义可以看到，这个函数的第一个参数是一个函数，剩下的参数是一个或多个序列，返回值是一个集合。

function可以理解为是**一个一对一或多对一函数**，map的作用是以参数序列中的每一个元素调用function函数，返回包含每次function函数返回值的list。

比如要对**一个序列**中的**每个元素**进行**平方运算**：

map(lambda x: x \*\* 2, [1, 2, 3, 4, 5])

返回结果为：

[1, 4, 9, 16, 25]

**在参数存在多个序列时**，会依次以每个序列中相同位置的元素做参数调用function函数。

比如要对两个序列中的元素依次求和。

map(lambda x, y: x + y, [1, 3, 5, 7, 9], [2, 4, 6, 8, 10])

map返回的list中第一个元素为，参数序列1的第一个元素加参数序列2中的第一个元素(1 + 2)，

list中的第二个元素为，参数序列1中的第二个元素加参数序列2中的第二个元素(3 + 4)，

依次类推，最后的返回结果为：

[3, 7, 11, 15, 19]

要注意function函数的参数数量，要和map中提供的集合数量相匹配。

**如果集合长度不相等，会以最小长度对所有集合进行截取。**

**当函数为None时**，操作和zip相似：类似def f(x,y): return (x,y)

map(None, [1, 3, 5, 7, 9], [2, 4, 6, 8, 10])

返回结果为：

[(1, 2), (3, 4), (5, 6), (7, 8), (9, 10)]

**2.filter函数**

filter函数会对指定序列执行过滤操作。

filter函数的定义：

filter(function or None, sequence) -> list, tuple, or string

function是一个谓词函数，接受一个参数，返回布尔值True或False。

filter函数会对序列参数sequence中的每个元素调用function函数，最后返回的结果包含调用结果为True的元素。

返回值的类型和参数sequence的类型相同

比如返回序列中的所有偶数：

def is\_even(x):

return x & 1 != 0

filter(is\_even, [1, 2, 3, 4, 5, 6, 7, 8, 9, 10])

返回结果为：

[1, 3, 5, 7, 9]

如果function参数为None，返回结果和sequence参数相同。

**3.reduce函数**

reduce函数，reduce函数会对参数序列中元素进行累积。

reduce函数的定义：

reduce(function, sequence[, initial]) -> value

function参数是一个有两个参数的函数，reduce依次从sequence中取一个元素，和上一次调用function的结果做参数再次调用function。

第一次调用function时，如果提供initial参数，会以sequence中的第一个元素和initial作为参数调用function，否则会以序列sequence中的前两个元素做参数调用function。

reduce(lambda x, y: x + y, [2, 3, 4, 5, 6], 1)

结果为21(  (((((1+2)+3)+4)+5)+6)  )

reduce(lambda x, y: x + y, [2, 3, 4, 5, 6])

结果为20