

匿名函数

1

匿名函数

——匿名函数

lambda

- python允许用lambda关键字创造匿名函数
- 匿名是因为不需要以标准的def方式来声明
- 一个完整的lambda “语句” 代表了一个表达式，这个表达式的定义体必须和声明放在同一行

```
lambda [arg1[, arg2, ... argN]]: expression
```

```
>>> a = lambda x, y: x + y
```

```
>>> print(a(3, 4))
```

```
7
```

filter()函数

- `filter(func, seq)`：调用一个布尔函数`func`来迭代遍历每个序列中的元素；返回一个使`func`返回值为`true`的元素的序列
- 如果布尔函数比较简单，直接使用`lambda`匿名函数就显得非常方便了

```
>>> data = filter(lambda x: x % 2, [num for num in range(10)])  
>>> print(data)      #过滤出10以内的奇数  
[1, 3, 5, 7, 9]
```

map()函数

- `map(func, seq1[,seq2...])`：将函数`func`作用于给定序列的每个元素，并用一个列表来提供返回值

```
>>> data = map(lambda x: x + 2, [num for num in range(5)])  
>>> print data          #将0到4每个数值加2，生成新的列表  
[2, 3, 4, 5, 6]
```

reduce()函数

- `reduce(func, seq[, init])`：将二元函数作用于seq序列的元素，每次携带一对（先前的结果以及下一个序列元素），连续的将现有的结果和下一个给值作用在获得的随后的结果上，最后减少序列为一个单一的返回值

```
>>> data = reduce(lambda x,y: x+y, range(1, 6)) #将1至5累加
>>> print(data)
15
```



更多精彩...



<http://bj.linux.tedu.cn/>
企业QQ：86198501