Python3-CookBook：53、找出当月的日期范围

作者：zhaolixiang
原文链接：https://ld246.com/article/1568364871496
来源网站：链滴

1 问题

我们有一些代码需要循环迭代当月中的每个日期，我们需要一种高效的方法来计算出日期的范围。

2 解决方案

对日期进行循环迭代并不需要事先构建一个包含所有日期的列表。只需计算出范围的开始和结束日期然后在迭代时利用datetime.timedelta对象来递增日期就可以了。

下面这个函数可接受任意的datetime对象，并返回一个包含本月第一天和下个月第一天日期的元组。例如下：

```
from datetime import datetime, date, timedelta
import calendar

def get_month_range(start_date=None):
    if start_date is None:
    start_date = date.today().replace(day=1)
_, days_in_month = calendar.monthrange(start_date.year, start_date.month)
end_date = start_date + timedelta(days=days_in_month)
return (start_date, end_date)
```

当准备好这个函数后，对日期范围做循环迭代就变得非常简单了：

```
>>> a_day = timedelta(days=1)
>>> first_day, last_day = get_month_range()
>>> while first_day < last_day:
...     print(first_day)
...     first_day += a_day
...
2012-08-01
2012-08-02
2012-08-03
2012-08-04
2012-08-05
2012-08-06
2012-08-07
2012-08-08
2012-08-09
#... and so on...
```

3 讨论

上面的代码首先计算出相应月份中第一天的日期。一种快速求解的方法是利用date或者datetime对的replace()方法，只要将属性days设为1就可以了。关于replace()方法，一个好的方面就是它创建出对象和我们的输入对象类型是一致的。因此，如果输入是一个date实例，那得到的结果也是date实例同样，如果输入是datetime实例，得到的也是datetime实例。

此外，我们用calendar.monthrange()函数来找出待求解的月份中有多少天。当需要得到有关日历方的基本信息时，calendar模块都会非常有用。monthrange()是其中唯一的一个可返回元组的函数，组中包含当月第一个工作日的日期[1]以及当月的天数（28～31）。

一旦知道了这个月中有多少天，那么结束日期就可以通过在起始日期上加上一个合适的timedelta对来表示。尽管很微不足道，但本节给出的解决方案中一个重要的方面就是结束日期并不包含在范围内因为它实际上是下个月的第一天）。这刚好应对了Python中切片和range操作的行为，这些操作永远会将结束点包含在内。

要循环迭代日期范围，我们这里采用了标准的算术以及比较操作符。比如，timedelta实例可用来递日期，而<操作符用来检查当前日期是否超过了结束日期。

最理想的方法是创建一个专门处理日期的函数，而且用法和Python内建的range()一样。幸运的是，生成器来实现这样一个函数真的是非常容易：

```
def date_range(start, stop, step):
    while start < stop:
        yield start
        start += step
```

下面是使用这个函数的示例：

```
>>> for d in date_range(datetime(2012, 9, 1), datetime(2012,10,1),
                        timedelta(hours=6)):
...     print(d)
...
2012-09-01 00:00:00
2012-09-01 06:00:00
2012-09-01 12:00:00
2012-09-01 18:00:00
2012-09-02 00:00:00
2012-09-02 06:00:00
...
>>>
```

这里要再一次说明，之所以上述实现会如此简单，一个很重要的原因就在于日期和时间可以通过标准算术和比较操作符来进行操作。