

## 线程不安全测试代码

原因

解决方式1

解决方式2

解决方式3

解决方式4

## 线程不安全测试代码

```
1 public static void main(String[] args) {
2
3     SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
4     for (int i = 0; i < 10; ++i) {
5         new Thread(new Runnable() {
6             @Override
7             public void run() {
8                 try {
9                     System.out.println(sdf.parse("2017-12-13 15:17:27"));
10                } catch (ParseException e) {
11                    e.printStackTrace();
12                }
13            }
14        }).start();
15    }
16
17
18 }
```

会抛出异常 java.lang.NumberFormatException: multiple points, 或者：转换后的值错误

### 原因

SimpleDateFormat继承与 DateFormat类，在DateFormat类定义了一个protected属性的 Calendar类的对象。  
在多线程环境下，这个Calendar属性会被改乱，从而导致错误

### 解决方式1

```
1 public class DateUtil {
2
3     public static String formatDate(Date date) throws ParseException{
4         SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
5         return sdf.format(date);
6     }
7
8     public static Date parse(String strDate) throws ParseException{
9         SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
10        return sdf.parse(strDate);
11    }
12 }
```

#### 解决方式2

```
1 public class DateSyncUtil {
2
3     private static SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
4
5     public static String formatDate(Date date) throws ParseException{
6         synchronized(sdf){
7             return sdf.format(date);
8         }
9     }
10
11     public static Date parse(String strDate) throws ParseException{
12         synchronized(sdf){
13             return sdf.parse(strDate);
14         }
15     }
16 }
```

#### 解决方式3

```
1 public class ConcurrentDateUtil {
2
3     private static ThreadLocal<DateFormat> threadLocal = new ThreadLocal<DateFormat>() {
4         @Override
5         protected DateFormat initialValue() {
6             return new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
7         }
8     };
9
10    public static Date parse(String dateStr) throws ParseException {
11        return threadLocal.get().parse(dateStr);
12    }
13
14    public static String format(Date date) {
15        return threadLocal.get().format(date);
16    }
17 }
```

#### 解决方式4

抛弃JDK，使用其他类库中的时间格式化类，Joda-Time类库对时间处理方式比较完美，建议使用