sleep和wait的区别是面试中一个非常常见的问题，因为从表象来看，好像sleep和wait都能使线程处于阻塞状态，但是却有着本质上的却别。这篇文章就来好好分析一下。

整体的区别其实是有四个：

1、sleep是线程中的方法，但是wait是Object中的方法。

2、sleep方法不会释放lock，但是wait会释放，而且会加入到等待队列中。

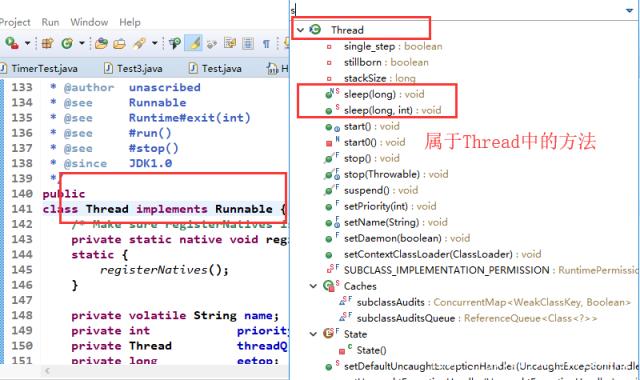
3、sleep方法不依赖于同步器synchronized，但是wait需要依赖synchronized关键字。

4、sleep不需要被唤醒（休眠之后推出阻塞），但是wait需要（不指定时间需要被别人中断）。

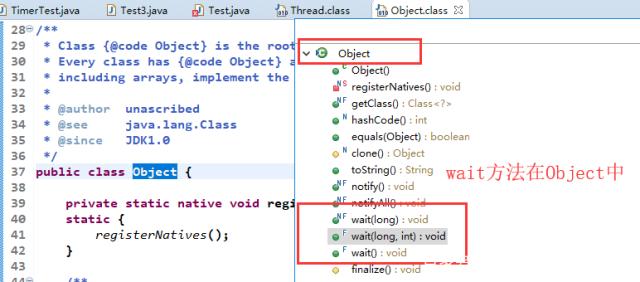
下面我们就根据这四个区别来分析。

**一、sleep是线程方法，wait是Object方法**

这个如何验证呢？我们还需要到jdk源码中看看。首先进入到Thread的源码中看一下，然后俺ctrl+O就可以查看方法列表。在最上面可以搜寻，我们输入“s”，就可以查看所有以s开头的方法了。



我们会发现，slee方法真实的在Thread线程类中。下面我们以同样的方法查看wait。



这是第一个区别很容易验证，下面我们来看第二个。

**二、sleep不释放lock，wait会释放**

这个如何验证呢？这就需要代码了。先看我们的sleep方法



我们看一下运行结果：

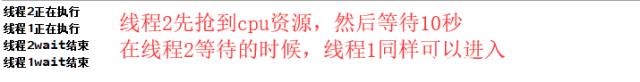


在上面的结果中，线程2先获取了cpu资源，然后开始执行休眠，在休眠过程中线程1是没法执行的，必须要等待线程2结束之后才可以。这也就是说sleep方法不会释放锁，让其他线程进来。

然后我们测试一下wait方法。



在上面的例子中，我们使用wait方法等待10秒钟，然后结束。我们看一下结果：



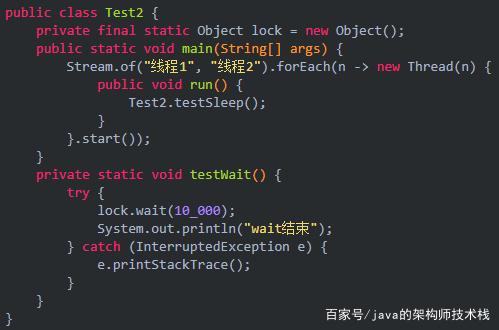
这个过程就验证了第二条区别，我们接下来看第三个。

**三、sleep不依赖同步方法，wait需要**

我们还是一次来验证。首先我们测试sleep方法。



这个方法会依次运行，不会出现任何异常。然后我们主要是看wait方法。



我们运行以下，看一下结果：



OK，下面我们验证一下第四条区别：

**四、sleep不需要被唤醒，wait需要**

sleep方法很简单，我们主要关注wait方法。看代码：

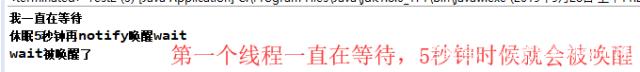
首先我们定义两个方法，一个等待方法，一个唤醒方法。



然后再去测试一下：



如果没有唤醒方法，那第一个线程就会处于一直等待的状态，第二个线程唤醒了之后就不再等待了。



以上就是四个区别的完整验证，如有问题还请指正。