

一个理解wait()与notify()的例子 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/252/2021_2022__E4_B8_80_E4_B8_AA_E7_90_86_E8_c104_252428.htm 下面是我原来

在CSDN论坛上看到的一个帖子，涉及到同步,wait(),notify()等概念的理解，我试着根据原来的一些回复和Think in Java上的相关概念将wait()和notify()这两个方法剖析了一下，欢迎指教。

问题如下：//分析这段程序，并解释一下，着重讲

讲synchronized、wait(),notify 谢谢！
class ThreadA { public static void main(String[] args) { ThreadB b=new ThreadB(). b.start().

System.out.println("b is start...."). synchronized(b)//括号里的b是什么意思,起什么作用? { try { System.out.println("Waiting for b to complete..."). b.wait().//这一句是什么意思，究竟让谁wait?

System.out.println("Completed.Now back to main thread"). }catch (InterruptedException e){} } System.out.println("Total is :" b.total).

}}class ThreadB extends Thread{ int total. public void run() { synchronized(this) { System.out.println("ThreadB is running.."). for (int i=0;i { total =i. System.out.println("total is " total). } notify(). } }}

要分析这个程序,首先要理解notify()和wait(),为什么在前几天纪录线程的时候没有纪录这两个方法呢,因为这两个方法本来就不属于Thread类,而是属于最底层的object基础类的,也就是说不光是Thread，每个对象都有notify和wait的功能，为什么

？因为他们是用来操纵锁的,而每个对象都有锁,锁是每个对象的基础,既然锁是基础的,那么操纵锁的方法当然也是最基础了.

再往下看之前呢,首先最好复习一下Think in Java的14.3.1中第3部分内容：等待和通知,也就是wait()和notify了. 按照Think

in Java中的解释:"wait()允许我们将线程置入“睡眠”状态,同时又“积极”地等待条件发生改变.而且只有在一个notify()或notifyAll()发生变化的时候,线程才会被唤醒,并检查条件是否有变."我们来解释一下这句话."wait()允许我们将线程置入“睡眠”状态",也就是说,wait也是让当前线程阻塞的,这一点和sleep或者suspend是相同的.那和sleep,suspend有什么区别呢?区别在于"(wait)同时又“积极”地等待条件发生改变",这一点很关键,sleep和suspend无法做到.因为我们有时候需要通过同步(synchronized)的帮助来防止线程之间的冲突,而一旦使用同步,就要锁定对象,也就是获取对象锁,其它要使用该对象锁的线程都只能排队等着,等到同步方法或者同步块里的程序全部运行完才有机会.在同步方法和同步块中,无论sleep()还是suspend()都不可能自己被调用的时候解除锁定,他们都霸占着正在使用的对象锁不放.而wait却可以,它可以让同步方法或者同步块暂时放弃对象锁,而将它暂时让给其它需要对象锁的人(这里应该是程序块,或线程)用,这意味着可在执行wait()期间调用线程对象中的其他同步方法!在其它情况下(sleep啊,suspend啊),这是不可能的.但是注意我前面说的,只是暂时放弃对象锁,暂时给其它线程使用,我wait所在的线程还是要把这个对象锁收回来的呀.wait什么?就是wait别人用完了还给我啊!好,那怎么把对象锁收回来呢?第一种方法,限定借出去的时间.在wait()中设置参数,比如wait(1000),以毫秒为单位,就表明我只借出去1秒中,一秒钟之后,我自动收回.第二种方法,让借出去的人通知我,他用完了,要还给我了.这时,我马上就收回来.哎,假如我设了1小时之后收回,别人只用了半小时就完了,那怎么办呢?靠!当然用完了就收回了,还管我设的是多长时间啊.那么

别人怎么通知我呢?相信大家都可以想到了,notify(),这就是最后一句话"而且只有在一个notify()或notifyAll()发生变化的时候,线程才会被唤醒"的意思了.因此,我们可将一个wait()和notify()置入任何同步方法或同步块内部,无论在那个类里是否准备进行涉及线程的处理。而且实际上,我们也只能在同步方法或者同步块里面调用wait()和notify().这个时候我们来解释上面的程序,简直是易如反掌了.synchronized(b){...};的意思是定义一个同步块,使用b作为资源锁。b.wait().的意思是临时释放锁,并阻塞当前线程,好让其他使用同一把锁的线程有机会执行,在这里要用同一把锁的就是b线程本身.这个线程在执行到一定地方后用notify()通知wait的线程,锁已经用完,待notify()所在的同步块运行完之后,wait所在的线程就可以继续执行. 100Test 下载频道开通,各类考试题目直接下载。详细请访问 www.100test.com