

基于JDK5.0的一些Thread总结 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/257/2021_2022__E5_9F_BA_E4_BA_8EJDK5_c67_257600.htm 在新的JDK5.0中，对thread做

了一些改进，我通过阅读一些资料写下了下面的总结，请大家看看。1.创建线程 在java中实现多线程有两种方法，一个是直接继承Thread类，一个是实现Runnable接口，但是推荐的是第二种。因为在逻辑上应该要把一个线程要做的事情以及做这个事情的方法分开；对于Thread来讲，它只负责线程的操作，而具体要做的事情就应该放在Runnable中。但不管是那种方式，都要实现public void run()方法，但启动线程用start而不是run。2.终止线程 在1.0中，可以用stop方法来终止，但是现在这种方法已经被禁用了，改用interrupt方法。interrupt方法并不是强制终止线程，它只能设置线程的interrupted状态，而在线程中一般使用一下方式： while

```
(!Thread.currentThread().isInterrupted() & more work to do){do more work} 而被block的线程在被调用interrupt时会产生InterruptedException，此时是否终止线程由本线程自己决定。
```

程序的一般形式是： public void run(){try{. .while

```
(!Thread.currentThread().isInterrupted() & more work to do){do more work}}catch(InterruptedException e){// thread was interrupted during sleep or wait}finally{cleanup, if required}// exiting the run method terminates the thread} Thread.sleep方法也会产生InterruptedException，因此，如果每次在做完一些工作后调用了sleep方法，那么就不用检查isInterrupted，而是直接捕捉InterruptedException。在捕捉到InterruptedException时，
```

如果没有其他的事情可做，最好做一下处理，不能用{}1)
void mySubTask(){...try { sleep(delay). }catch
(InterruptedException e) { Thread().currentThread().interrupt(). }.
..}或是2) void mySubTask() throws InterruptedException{..
.sleep(delay)...} 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