

使用实时Java降低Java应用程序的易变性(2)Java认证考试 PDF
转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/644/2021_2022__E4_BD_BF_E7_94_A8_E5_AE_9E_E6_c104_644691.htm

Java 服务器示例 在本文剩余部分，我们将应用在前面章节中介绍的一些想法，使用 Java 类库中的 Executors 服务构建一个相对简单的 Java 服务器应用程序。只需少量应用程序代码，Executors 服务就可以用于创建一个服务器来管理工作者线程池，如清单 5 所示：

：清单 5. 使用 Executors 服务的 Server 和 TaskHandler 类

```
import java.util.concurrent.Executors. import
java.util.concurrent.ExecutorService. import
java.util.concurrent.ThreadFactory. class Server { private
ExecutorService threadPool. Server(int numThreads) {
ThreadFactory theFactory = new ThreadFactory(). this.threadPool
= Executors.newFixedThreadPool(numThreads, theFactory). }
public void start() { while (true) { // main server handling loop, find
a task to do // create a "TaskHandler" object to complete this
operation TaskHandler task = new TaskHandler().
this.threadPool.execute(task). } this.threadPool.shutdown(). } public
static void main(String[] args) { int serverThreads =
Integer.parseInt(args[0]). Server theServer = new
Server(serverThreads). theServer.start(). } } class TaskHandler
extends Runnable { public void run() { // code to handle a "task" } }
```

此服务器可以创建所有需要的线程，直到达到创建服务器（从此示例中的命令行解码）时指定的最大数量。每个工作者线程使用 TaskHandler 类执行一部分工作。出于我们的目的，

我们将创建一个 TaskHandler.run() 方法，它每次运行都应该花相同的时间。因此，执行 TaskHandler.run() 的时间上的任何易变性都源自于底层 JVM 中的暂停或易变性、某个线程问题或在堆栈的较低级别上引入的暂停。清单 6 给出了

TaskHandler 类：清单 6. 具有可预测性能的 TaskHandler 类

```
import java.lang.Runnable; class TaskHandler implements Runnable
{ static public int N=50000; static public int M=100; static long
result=0L; 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
```