

使用java.util.concurrent实现的线程池、消息队列功能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\\_A8java\\_c104\\_644877.htm](https://www.100test.com/kao_ti2020/644/2021_2022__E4_BD_BF_E7_94_A8java_c104_644877.htm)

ThreadPoolManager类：负责管理线程池，调用轮询的线程来访问字符串缓冲区的内容，维护缓冲区，当线程池溢出时抛出的Runnable任务被加入到字符串缓冲区。

```
public class ThreadPoolManager { private static
ThreadPoolManager tpm = new ThreadPoolManager(). // 线程池
维护线程的最少数量 private final static int CORE_POOL_SIZE =
4. // 线程池维护线程的最大数量 private final static int
MAX_POOL_SIZE = 10. // 线程池维护线程所允许的空闲时间
private final static int KEEP_ALIVE_TIME = 0. // 线程池所使用的
缓冲队列大小 private final static int WORK_QUEUE_SIZE = 10.
// 消息缓冲队列 Queue msgQueue = new LinkedList(). // 访问消息
缓存的调度线程 final Runnable accessBufferThread = new
Runnable() { public void run() { // 查看是否有待定请求，如果有，
则创建一个新的AccessDBThread，并添加到线程池中 if(
hasMoreAcquire() ) { String msg = ( String ) msgQueue.poll().
Runnable task = new AccessDBThread( msg ). threadPool.execute(
task ). } } }. 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
```