怒放的生命

永久域名 http://muscle-liu.javaeye.com



muscle-liu

浏览: 29359 次

性别: 💣

来自: 广州



详细资料

留言簿

搜索本博客

最近访客 客

>>更多访





zouyusu

fdx123





shengyuan36(wbhltcx

博客分类

hsql 初试之 exception | SingleThread,MutiThread and Thread Pool(...

1顶 0踩

2009-02-13

对 ThreadPool 的理解

文章分类:Java编程

虽然从 Java5 开始 JDK 里的 java.util.concurrent 包内建了线程池, 你不必自己实现线程池, 但理解线程的实现原理对 Java 编程很有 用。

当你想把运行在你的程序中的线程控制在一定的数量之内,线程池就显得非常有用。

引用

原理:

用一个阻塞队列里(Blocking Queue)来存储线程池的所有空闲线程。不用为每个任务都创建一个新的线程,可以把任务当参数传到 线程池里。只要当线程池有空闲的线程时,这个任务就会被执行。

BlockQueue.java

Java代码

```
public class BlockingQueue {
 1.
 2.
 3.
          private List queue = new LinkedList();
          private int limit = 10;
 4.
          public BlockingQueue(int limit) {
 6.
               this.limit = limit;
 8.
          }
 9.
          public synchronized Object enqueue(Object item)
10.
                   throws InterruptedException {
11.
12.
               while (this.queue.size() == this.limit) {
                   System.out.println("Arrive to the pool's max size:"+this.limit);
13.
14.
                   wait();
```

```
■ 全部博客 (37)
```

■ <u>java基础 (9)</u>

■ <u>j2me (11)</u>

■ web应用 (3)

■ 随想随写 (6)

C & Solaris (6)

Flex (3)

我的留言簿 >> 更多留言

■ muscle 哥, 你好你好

-- by endeavor416

其他分类

■ <u>我的收藏</u> (10)

■ <u>我的论坛主题贴</u> (39)

■ 我的所有论坛贴 (26)

我的精华良好贴 (0)

最近加入圈子

■ <u>广州JavaEyer饭局群</u>

存档

2010-01 (1)

2009-09 (1)

2009-08 (1)

■ 更多存档...

最新评论

■ <u>项目进入 UAT 阶段,不那 ...</u> liucheng?

-- by <u>vk0805</u>

■ Unix C 延时函数小结

```
15.
               if (this.queue.size() == 0) {
16.
                   notifyAll();
17.
18.
               }
19.
               System.out.println("Add a new thread to the pool.Pool's size before
20.
21.
22.
      adding:"+this.queue.size());
               this.queue.add(item);
23.
24.
               return this.queue.get(this.queue.size()-1);
25.
          }
26.
27.
           public synchronized Object dequeue()
28.
29.
                   throws InterruptedException {
               while (this.queue.size() == 0) {
30.
                   System.out.println("Pool's size is 0.");
31.
32.
                   wait();
33.
               }
34.
               if (this.queue.size() == this.limit) {
                   notifyAll();
35.
36.
               }
37.
38.
               System.out.println("Remove a thread from the pool.Pool's size before
39.
      removing:"+(this.queue.size()-1));
40.
               return this.queue.remove(0);
41.
          }
42.
43.
          public synchronized int size(){
               return this.queue.size();
45.
46.
          }
47.
```

当阻塞队列没达到界限值(最大值与最小值)时,插入与出列正常,没限制。阻塞队列达到最大值时,再想插入一个线程,队列就会停止操作,让插入在等待,同时唤醒出列操作。相反,当全部出列后,再想出列时,出列操作就会停止,在等待,同时唤醒插入操作。

线程池通常用在多线程服务端的,下边是实现线程池的其他类,其中:

Java代码

```
    /**
    * Implements the thread pool
```

```
3.
                                         * @author winxp
  唉最近遇到一个定时问题, 起初
                                  4.
  是想用两个线程解决, 但是越搞
                                  5.
                                       public class ThreadPool {
                                  6.
  越复杂,最后sleep解决...
                                  7.
                                           private BlockingQueue taskQueue = null;
  -- by jinleileiking
                                           private List<TaskThread> threads = new ArrayList<TaskThread>();
                                  8.
                                            private boolean isStopped = false;
                                  9.
■ JAVA对象比较器Comparator
                                 10.
  字符串,Integer 等实现了可比较
                                           /**
                                 11.
  接口Compareable,比较没那么
                                 12.
                                             * Create a new ThreadPool
                                             * @param noOfThreads Initiate num of threads in the thread pool线程池的初始化线程数
                                 13.
  麻烦吧...
                                             * @param maxNoOfTasks Max num of threads in the thread pool线程池的最大线程数
                                 14.
  -- by taoshui123
                                 15.
                                           public ThreadPool(int noOfThreads, int maxNoOfTasks) {
■ JAVA对象比较器Comparator
                                 16.
                                                taskQueue = new BlockingQueue(maxNoOfTasks);
                                 17.
  刚才做了javaeye规则测试题,
                                 18.
  差点没缓过神来~~~写的挺好的
                                 19.
                                               for (int i = 0; i < noOfThreads; i++) {</pre>
                                                   threads.add(new TaskThread(taskQueue));
                                 20.
  正需要 收藏了
                                 21.
  -- by mxdba321123
                                               for (TaskThread thread : threads) {
                                 22.
                                 23.
                                                    thread.start();
■ Java 思想写 Flex 的危险
                                 24.
                                               }
  很好奇楼主的代码, 那样的能跑
                                 25.
                                           }
  通不? 好像as里, 更习惯于:
                                 26.
                                 27.
  private var ...
                                 28.
  -- by <u>lydawen</u>
                                             * @param task
                                 29.
                                 30.
 评论排行榜
                                            public synchronized void execute(Runnable task){
                                 31.
                                               if(this.isStopped)throw
                                 32.
■ Java 思想写 Flex 的危险
                                 33.
                                                       new IllegalStateException("ThreadPool is stopped");
■ 项目进入 UAT 阶段,不那么忙
                                 34.
                                               try {
                                 35.
                                                   this.taskQueue.enqueue(task);
  了
                                               } catch (InterruptedException ex) {
                                 36.
■ Unix C 延时函数小结
                                                   ex.printStackTrace();
                                 37.
                                 38.
■ Pro*C 简单实现 Oracle 数据连
                                 39.
                                           }
  接池
                                 40.
                                            /**
                                 41.
■ 要写 C Program 啦
                                 42.
                                 43.
  ₹ RSS
                                           public synchronized void stop() {
                                 44.
  Add to Google
                                               this.isStopped = true;
                                 45.
```

[什么是RSS?]

```
for (TaskThread thread : threads) {
46.
47.
                   thread.interrupt();
48.
          }
49.
50.
      }
51.
52.
53.
       * Simulate that dequeue from the BlockingQueue, and execute the dequeuing task
54.
55.
      public class TaskThread extends Thread {
56.
          private BlockingQueue taskQueue = null;
57.
58.
          private boolean isStopped = false;
59.
60.
          public TaskThread(BlockingQueue queue) {
61.
               taskQueue = queue;
62.
          }
63.
          @Override
64.
65.
          public void run() {
               while (!isStopped()) {
66.
67.
                   try {
                       Runnable runnable = (Runnable) taskQueue.dequeue();
68.
                       runnable.run();
69.
70.
                   } catch (Exception e) {
                       //log or otherwise report exception,
71.
72.
                       //but keep pool thread alive.
73.
74.
75.
          }
76.
          @Override
77.
78.
          public synchronized void interrupt() {
79.
               isStopped = true;
80.
               super.interrupt();//break pool thread out of dequeue() call.
          }
81.
82.
          public synchronized boolean isStopped() {
83.
               return isStopped;
84.
85.
          }
86.
      }
```

测试类:

Java代码

```
/**
 1.
 2.
       * Testing class
 3.
      class TestThread extends Thread{
 4.
 5.
           private int num;
           public TestThread(int num){
 6.
               this.num = num;
 8.
          }
 9.
10.
           @Override
           public void run(){
11.
               System.out.println("TestThread index: "+num);
12.
13.
               try {
                   Thread.sleep(1000);
14.
15.
               } catch (InterruptedException ex) {
16.
17.
          }
18.
19.
           /**
20.
            * @param args
21.
22.
           public static void main(String[] args){
23.
               ThreadPool tp = new ThreadPool(3,10);
24.
25.
               for(int i = 0;i<15;i++){</pre>
                   TestThread test = new TestThread(i);
26.
27.
                   tp.execute(test);
28.
29.
          }
30.
```

学了Java,还找不到工作?

为何学苹果手机开发工资高一倍? 名企委托培养,本月报名学费**75**折... iphone.peixun.it

Google 提供的广告

0
 顶

hsql 初试之 exception | SingleThread,MutiThread and Thread Pool(...

	表情	图标		字体颜色: □[□ 字体大	:小: 🗆 🗆	对齐: □□		
9	9	<u> </u>		提示:选择您	活要装饰的文	文字, 按上列按钮即	可添加上相应的标	签	
3		<u>60</u>	-						
9	(ii)	(3)	3						
	9		-Do						

您还没有登录,请登录后发表评论(快捷键 Alt+S / Ctrl+Enter)