Java 中的线程：对象锁 - IV

切坦·库达尔卡

2021 年 6 月 17 日 警察

这是我之前关于“Java 中的线程：对象锁-III”的文章的延续

本系列文章试图帮助读者了解 Java 中的锁概念。假设读者对 Java 和 Java 中的线程创建有一些先验知识。

介绍

到目前为止，在“Java中的线程：对象锁”系列文章中，我们已经看到对象锁如何同步从属于同一类的对象创建的线程，例如线程`t1`和`t2`创建的线程以及`p1`对象锁如何同步创建的线程出两个不同的对象的，但属于同一类，例如，线程`t1`和`t2`处理器对象的创作出来`p1`并`p2`分别。

在本文中，我们将看到对象锁如何帮助同步由属于完全不同类的两个不同对象创建的线程。为了证明这一点，我们将有两个不同的类，即`processor`类和`memory`类。采取以下程序：

爪哇 缩小▲ 复制代码

```
locks.java
-----------

class processor implements Runnable{
    Object objLock;
    public processor(Object objLock){
        this.objLock = objLock;
    }
    public void run() {
        executeInstructions();
    }
    public void executeInstructions() {
        synchronized(objLock) {
            for (int i = 0 ; i<= 5; i++) {
                System.out.println("i = " + i + " In thread executeInstructions
                                    of processor class" + Thread.currentThread());
            }
        }
    }
}

class memory implements Runnable{
    Object objLock;
    public memory(Object objLock){
        this.objLock = objLock;
    }
    public void run() {
```

```
        checkMemory();
    }
    public void checkMemory() {
        synchronized(objLock) {
            for (int i = 0 ; i<= 5; i++) {
                System.out.println("i = " + i + " In thread checkMemory
                                    of memory class" + Thread.currentThread());
            }
        }
    }
}

class locks {
    public static void main(String[] args) {
        Object objLock = new Object();
        processor p1 = new processor(objLock);
        memory m1 = new memory(objLock);
        Thread t1 = new Thread(p1, "t1");
        Thread t2 = new Thread(m1, "t2");
        t1.start();
        t2.start();
    }
}

Output
======
i = 0 In thread executeInstructions of processor classThread[t1,5,main]
i = 1 In thread executeInstructions of processor classThread[t1,5,main]
i = 2 In thread executeInstructions of processor classThread[t1,5,main]
i = 3 In thread executeInstructions of processor classThread[t1,5,main]
i = 4 In thread executeInstructions of processor classThread[t1,5,main]
i = 5 In thread executeInstructions of processor classThread[t1,5,main]
i = 0 In thread checkMemory of memory classThread[t2,5,main]
i = 1 In thread checkMemory of memory classThread[t2,5,main]
i = 2 In thread checkMemory of memory classThread[t2,5,main]
i = 3 In thread checkMemory of memory classThread[t2,5,main]
i = 4 In thread checkMemory of memory classThread[t2,5,main]
i = 5 In thread checkMemory of memory classThread[t2,5,main]
```

上述程序编译并成功运行以产生所需的预期输出。如前所述，我们有两个类，即 vizprocessor和memoryclass。这两个类都实现了Runnable接口，这意味着我们将从这两个类创建线程。类处理器executeInstruction()从run方法调用，而类内存checkMemory()从run方法调用。这两种方法即。executeInstruction()并且checkMemory()有一个同步块，每个块都同步一个且只有一个，objLock因为这是我们在内部创建内存对象m1和处理器对象时通过构造函数传递给每个对象的对象。p1main()

我们t1从p1[类处理器的]处理器对象和[类内存]t2的内存对象外创建线程m1。接下来，t1和t2启动。t1获得时间片的那一刻，它获得了一个锁定objLock并开始在executeInstructions(). 由于t1拥有 上的锁objLock，t2必须等到t1释放锁。一旦t1完成 for 循环的执行，它就会释放objLock. 现在t2接管并完成。

通过这种方式，我们能够同步同步线程t1，并t2认为是两个不同的创造出object小号即。p1并且m1属于完全不同的类，即。processor和memory分别。

兴趣点

如果两个或多个线程只有一个且只有一个object公共锁，它们将同步。

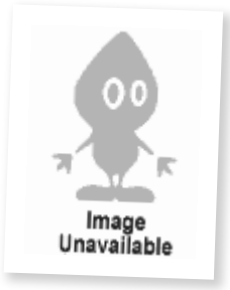
历史

- 2021 年 6 月17日：初步修订

执照

分享

关于作者



切坦·库达尔卡
软件开发人员（高级）
印度
手表
该会员

我是一名软件工程师，拥有大约 7 年以上的经验。我的大部分经验是在存储技术方面。

评论和讨论

添加评论或问题 电子邮件提醒 Search Comments

第一 页上一页 下一页

你的系列
Sean Ewington **18-Jun-21 21:34**

回复：你的系列
Chetan Kudalkar 19-Jun-21 18:09

回复：你的系列
Sean Ewington 21-Jun-21 20:49

刷新 1

一般 新闻 建议 问题 错误 答案 笑话 赞美 咆哮 管理员

使用Ctrl+Left/Right 切换消息，Ctrl+Up/Down 切换主题，Ctrl+Shift+Left/Right 切换页面。