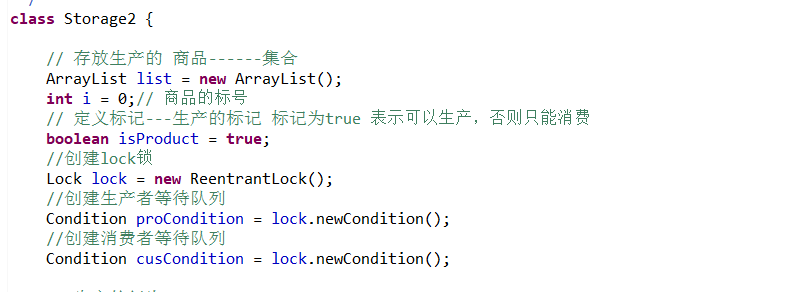
**使用lock和定义队列改进线程并发管理**



使用Lock lock = new ReentrantLock();创建一个lock锁，然后使用lock.newCondition()的方法创建两个队列，一个代表生产者，一个代表消费者队列。



首先先获取锁，然后使用try和finally方法，try方法里面放生产行为，也就是执行代码，finally里放释放锁lock.unlock（）。因为lock锁必须手动释放，如果不释放，会造成死锁的现象，先用while条件判断是否能生产，如果不能生产，使用生产者队列名.await方法将当前生产线程放进生产者队列当中，如果判断条件通过，就进入下方代码执行，然后生产完之后使用消费者队列名.signal（）方法唤醒消费者等待队列，因为队列是先进先出的功能，所以每次释放的都是第一个等待的消费者线程，也就优化了抢占资源的那一部分时间



同理，消费者也跟生产者一样，不过消费者放进的是消费者队列，消费者唤醒的是生产者队列，因为只有生产完了才能消费，消费完了才能生产。