**线程锁**

1. volatile仅能使用在变量级别,  
   synchronized则可以使用在变量,方法.
2. volatile仅能实现变量的修改可见性,但不具备原子特性，可以多个线程同时访问。  
   synchronized则可以保证变量的修改可见性和原子性，同一时刻只能有一个线程访问。
3. volatile不会造成线程的阻塞  
   synchronized可能会造成线程的阻塞.
4. volatile标记的变量不会被编译器优化，被volatile修饰的变量的读写操作都必须在内存中进行  
   synchronized标记的变量可以被编译器优化.
5. Synchronized和Lock的区别：

参考：<https://www.cnblogs.com/handsomeye/p/5999362.html>

1. Lock可以设置锁等待的时间、可以响应中断、允许多个线程进行读操作和告诉线程是否成功获得锁，synchronized则不能；
2. Lock不是Java语言内置的，它是接口，通过实现这个接口的类来实现同步访问；而synchronized是Java语言的关键字，属于内置特性。
3. Lock需要用户手动释放锁（即便发生异常也需要手动释放锁），而synchronized由系统自动释放；