

相同点：

Lock能完成synchronized所实现的所有功能。

不同：

1. ReentrantLock功能性更加全面，在多个条件变量和高度竞争锁的地方，用ReentrantLock更合适。
2. ReentrantLock必须在finally中释放锁，否则后果比较严重，编码角度synchronized更加简单点。
3. ReentrantLock提供了可轮询的锁请求，可以尝试去取得锁，如果取得成功则继续处理，取得不成功，可以等下次运行的时候处理，所以不容易死锁，而synchronized则一旦进入锁请求要么成功，要么一直阻塞，所以更容易产生死锁。

ReentrantLock获取锁定的三种方式。

1. lock()，如果获取了锁立即返回，如果别的线程持有锁，当前线程则一直处于休眠状态，直到获取锁
2. tryLock()，如果获取了锁立即返回true，如果别的线程正持有锁，立即返回false。
3. tryLock(long timeout, TimeUnit unit)，如果获取了锁，立即返回ture，如果别的线程正持有锁，会等待参数给定的时间，在等待的过程中，如果获取了锁，就返回true，如果等待超时，返回false。