

- synchronized是隐式锁，看不到锁了哪个对象，也看不到锁的开始和结束
- Lock是显式锁，Lock接口在JUC包下
- ReentrantLock类（可重入锁）实现了Lock接口，它拥有和synchronized相同的并发性和内存语义，在线程安全中更常用ReentrantLock来显示加锁和释放锁
- 用法：
 - 在线程类中创建一个private final ReentrantLock对象
 - run方法中，用try{}finally{}代码块将可能发生并发问题的代码括起来，代码前调用reentrantlock.lock()方法加锁放在try代码块中，finally代码块中调用reentrantLock.unlock方法解锁
 - 别忘记释放锁了
- ReentrantLock和synchronized的区别
 - ReentrantLock是显式锁，要手动开启和关闭锁，别忘记关闭了，synchronized是隐式锁，出了代码块作用域自动释放锁
 - ReentrantLock只作用于代码块，synchronized可以作用于代码块和方法
 - 使用Lock锁，jvm将花费较少的时间来调度线程，效率更好，且具有更好的可拓展性（有更多的子类）
 - 优选使用顺序：Lock > 同步块 > 同步方法