

线程安全问题：

- 1.将成员变量用static 修饰为静态变量。
- 2.调用Thread的构造方法。

线程安全问题产生的原因：

- 1.多个线程在同时操作共享的数据
- 2.操作共享数据的线程代码有多个

当一个线程在执行操作共享数据的多条代码的过程中，其他线程参与了运算，就会导致线程安全问题。

线程安全问题的解决方式：

- 1.synchronized(对象){需要被同步的代码块}
- 2.用synchronized去修饰方法（函数）
- 3.Java1.5后增加的Lock锁

synchronized相应的操作：

wait();线程等待

notify();线程唤醒（线程池中任意一个）

notifyAll();线程池中等待的线程全部唤醒

Lock相应的方法（操作）：

Lock锁 一个lock锁可以挂多个监视器。

lock.lock() 获取锁

lock.unlock(); 释放锁

Lock lock =new ReentrantLock();

Conditon 监视器=lock.newCondition();

Condition的方法：

await(); 线程等待

signal(); 线程唤醒

signalAll(); 线程唤醒