volatile定义的东西

并不能保存原子性，但是能保证内存的可见性（多线程看到的是一样的）和有序性

并且可以让jvm不优化相关对象，

可以解决，double check

解决单例，懒加载

推荐 通过内部类的使用加载 原因参考文件

**package** com.wangyunpeng.concurrent.chapter1;

**public class** SingletonObject6 {

**private** SingletonObject6(){

}

*// static 在线程运行的时候只会被加载一次，而且严格的线程同步*

*// 这个内部静态类会在程序第一次的调用这个类的时候主动加载解决了效率问题*

**private static class** InstanceHolder{

**private final static** SingletonObject6 ***instance*** = **new** SingletonObject6();

}

**public static** SingletonObject6 getInstance(){

**return** InstanceHolder.***instance***;

}

}