1.利用volatile的可见性，用它作为状态标记

volatile boolean start = true；

Thread-1

while（start） {

code

}

Thread-2

void close(){

start = false;

}

保证在上一个线程可以被发现

2.保证屏蔽前后的一致性

volatile boolean init；

----Thread1----

obj = createObj()

init = true

----Thread2----

while(!init){

sleep();

}

useTheObj(obj);

就算上面的代码中两个线程的部分不具有因果性，但是有volatile修饰之后的代码块能保证init前后顺序的有序性