TestThread1

1. 定义Runner类，重写run方法，用循环输出运行Runner类时的状态
2. 传入一个Runner类来新建Thread类，调用Thread类的start方法
3. 写主程序的运行状态（循环）

TestInterrupt

1. util包 ，继承Thread类创建MyThread类，重写run方法（不能抛出异常），用死循环每隔一秒打印时间，当有异常时停止
2. 主线程沉睡10秒后捕捉异常打断子线程

TestSync

1. 公共类Testsync实现Runnable接口且传给Thread线程t1和t2（用setName()设定好两个线程的名字），主方法外new一个Timer类，重写run方法，在run方法中用currentThread()和getName()方法以及Time类的自己写的add方法打印出此线程是第几个调用timer的线程
2. 写Timer类，写好static变量num，写add方法，并用sleep(1000)使得打印结果更明显。

ProducerConsumer

1. 定义一个窝头类，用数字编号来构造，且重写toString方法，便于打印
2. 定义一个框子类SyncStack，内有编号index，和一个含有6个窝头的窝头数组，定义同步的放入窝头的方法push，用while关键字而不是if语句来判断是否筐子里已经塞满了窝窝头，若塞满，则等待并捕获异常，若没有塞满则跳出循环并通知其他等待的线程，最后将此窝头放入将标记号+1。定义同步的拿出窝头的方法pop，判断是否筐子里已经没有窝头，若没有窝头，则等待，若有窝头则通知其他等待的线程，并将编号减一得到最上面的窝头并返回此窝头。
3. 定义生产者类Producer，告诉他把生产出来的窝头扔去哪个框（即构造方法），重写run方法，用for循环生产20个窝头并放入框子中，且每放一个就打印一次，且sleep随机的时间。消费者与此类似。
4. 主方法造框，并生出一个消费者和生产者，并令其成为两个线程。最后为了看到notify的效果，可以让pop的出栈时间加快（加快消费）。