2017/3/13 22:36

调试上：发现自己的mes项目坏掉了，在extendobject实体类上花了太多的时间，最后还是重新下载了一个项目，明天准备把这个项目的安装过程整理成一个详细的文档。

多线程：今天主要学习了几个主要的方法，印象深刻的有interrupt和isalive以及run方法执行的线程是main，了解了线程不安全是怎么回事，查询了怎么使用共享变量的三种方法，但是还是不能灵活运用。

2017/3/14 22:51

调试上：今天上午做了一上午的实验，中午又接到了学校财务的电话，很受伤，烦的一批，中午也没有好好学习，一直在毫无目的的浏览网页，下午上了两节没什么用的课，然后就去银行解决问题，也没有调试，晚上回来就去上课去了，很累。

多线程：10点才开始看书，只学习到了两点：如何中断线程，在线程中加入循环，如果遇到中断就跳出循环，但是实际上是没有中断线程，因为如果下面还有for循环外面还有语句，仍然会执行。所以可以抛出异常，然后就不会执行了，直接catch，然后中断。接下来就是在睡眠中中断，如果线程正在睡眠中断了，那么就会将sleep的状态清空执行剩下的语句，然后catch。如果是先中断，那么遇到sleep就会中断

2017/3/15 22:33

调试上：虽然白天起床时间都比较晚，但是很投入，整理出了不少问题，但是也学到了一个道理，遇到事情不要马上问别人，自己需要查询错误的原因，也能学到不少东西。

多线程：晚上又没有好好学习，只看了一点，将睡眠中中断学到了，先睡眠和后睡眠有什么区别，怎么让线程暂停，suspend虽然过时了，但是会占用资源，stop也是过时的，会抛出ThreadDeath异常。Yield方法会放弃资源，但是时间不确定。

2017/3/16 22:35

调试上：没起什么风波

学习上：高老师的课上看了会多线程，知道了守护线程的概念，了解到局部变量是在每个线程的栈中，因此不会出现非线程安全，所以局部变量线程安全，实例变量非线程安全。为了保证线程安全，在共享方法上加入synchronized保证线程安全，同一个对象只有一把锁，如果一个对象有两个synchronized的方法需要等待线程结束。

2017/3/19 10:39

调试上：早上解决了一个大问题，心情舒畅

学习上：昨天晚上看了一会多线程，主要内容包括：同步锁，在当前对象调用了一个同步方法，获取当前对象的锁，然后在该方法内又调用了该对象的另一个方法，那么可以直接进入该方法，这就是同步锁。当然父子继承也有同步锁，子类的同步方法调用父类的同步方法，可以直接进入到该方法。

2017/3/20 8:41

调试上：解决了一个最大的问题，现在没什么事，闲下来有时间了。

学习上：昨天花了相对多一点的时间去学习多线程，主要内容包括：同步线程的弊端，有时会让其他的线程等待较长的时间。如果用synchronized修饰静态方法，那么锁住的是整个class类，而不是一个实例对象。如果同步方法中遇到异常将自动释放锁。