### JAVA第五阶段—DAY02-JAVA作业答案

1. 编程题：
2. 用三种方式实现线程
3. 用同步代码块、同步方法、同步锁来实现转账的线程同步
4. 等待和通知实现生产者和消费者
5. 阻塞队列实现生产者和消费者
6. 面试题：
7. 进程和线程的区别

一个进程包含一个或多个线程

每个进程都有自己独立的内存空间，线程没有自己独立的内存空间，线程共享所在进程的内存空间

进程是重量级的单元，需要系统资源比较多，线程是轻量级单元，需要资源比较少

1. 线程有哪几种实现方式

继承Thread、实现Runnable

推荐实现Runnable方式，因为实现接口不影响类继承其它类

1. 线程的生命周期有哪些

1. 新建

线程new出来，没有调用start方法，CPU没有分配时间片

2. 就绪

线程调用了start方法，CPU准备分配时间片，但是没有真正分配

3. 运行

线程抢到了CPU，开始执行run方法

4. 阻塞

遇到特殊情况会进入阻塞状态，阻塞状态结束后，回到就绪状态

5. 死亡,run方法执行完毕后线程进入死亡状态

1. 线程同步有哪些实现

同步方法、同步代码块、同步锁

1. 性能：同步锁 > 同步块 > 同步方法

2. 同步锁提供了大量的方法，也可以if配合使用，更加灵活

1. ThreadLocal是什么

ThreadLocal是实现线程同步的另外一种解决方案。

hreadLocal是以空间换时间，每个线程拥有资源的一个副本，不会相互影响。

1. 乐观锁和悲观锁的区别

- 悲观锁

想法悲观，认为当前的资源存在竞争，所以每次获得资源时都会上锁，阻塞住其它线程。

数据库中的行锁、表锁、读锁、写锁都属于悲观锁，Java的synchronized和ReentrantLock也属于悲观锁。

悲观锁会降低系统性能和吞吐量，提高数据的安全性，适用于多写少读的场景。

- 乐观锁

想法乐观，认为当前的资源不存在竞争，所以每次获得资源时都不上锁

乐观锁执行效率高，有利于提高系统吞吐量，适用于多读少写的场景。

1. 生产者和消费者是什么

一种多线程开发的设计模式，在多线程开发当中，如果生产者处理速度很快，而消费者处理速度很慢，那么生产者就必须等待消费者处理完，才能继续生产数据。同样的道理，如果消费者的处理能力大于生产者，那么消费者就必须等待生产者。为了解决这种生产消费能力不均衡的问题，所以便有了生产者和消费者模式。

8）线程池是什么

线程池的作用是：首先会在池中分配一定数量的线程，线程使用完后会回到池中，等待下一个任务，线程资源就得到回收利用，减少服务器资源的消耗，提高了性能。