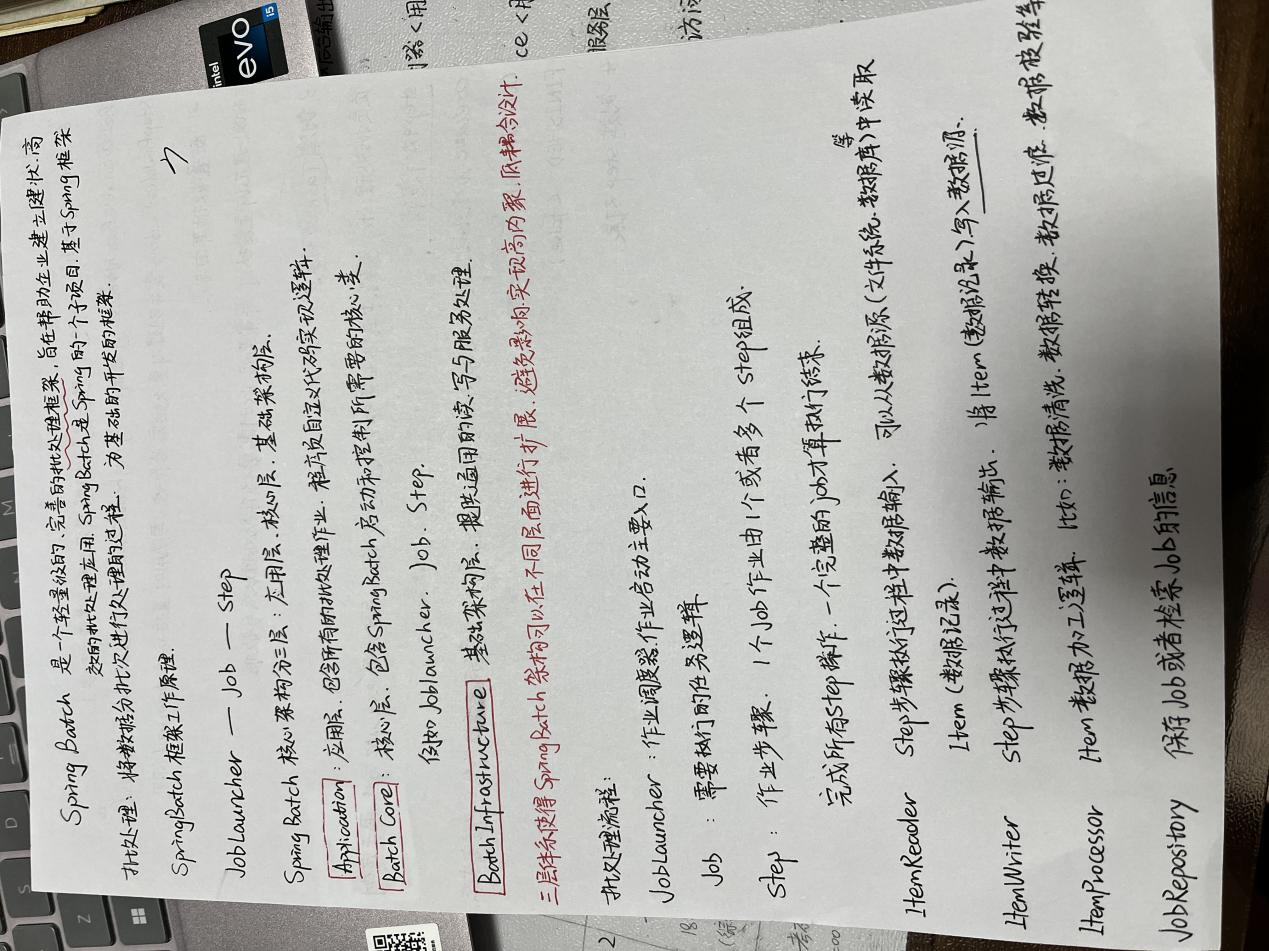
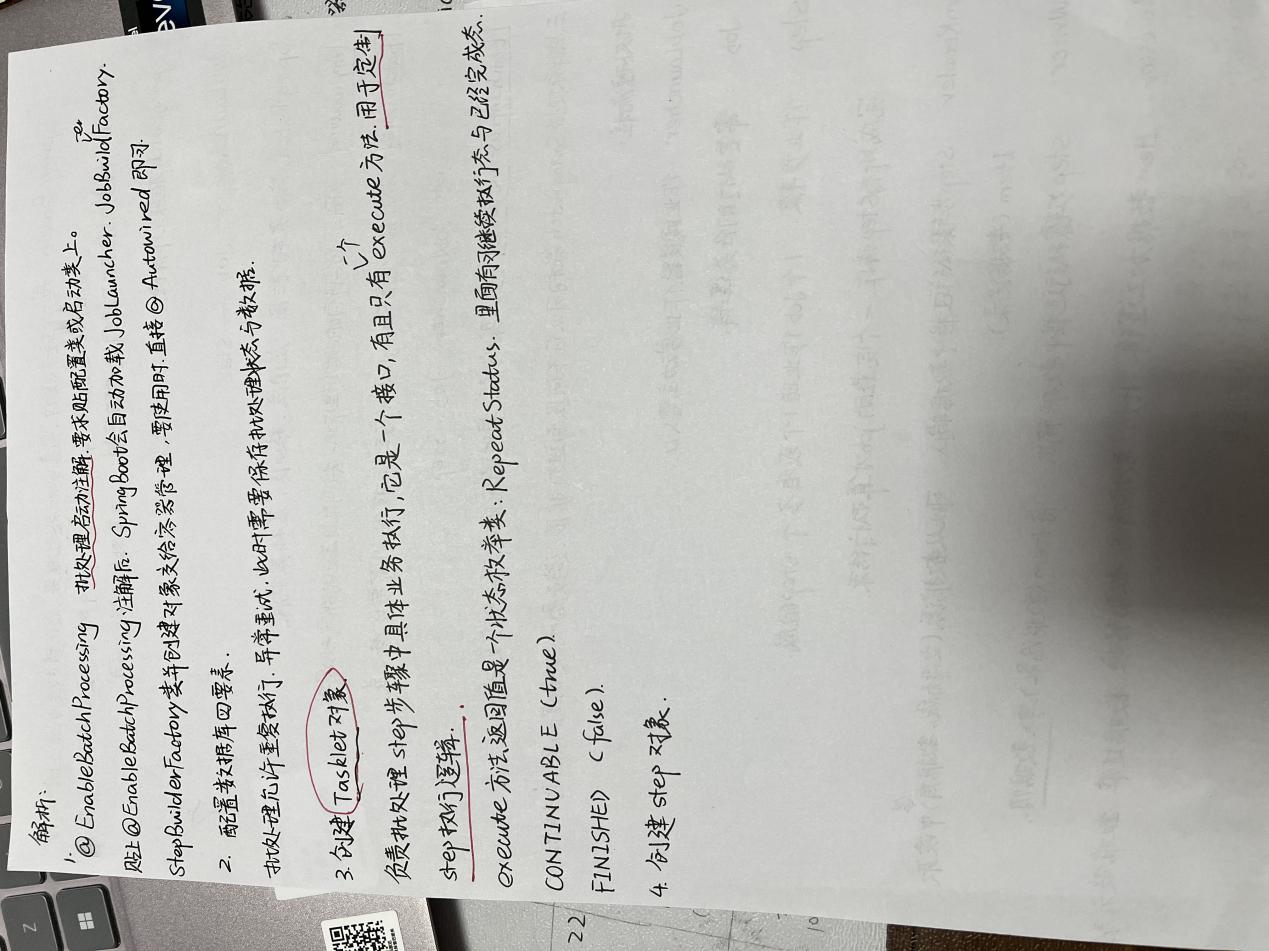
**SpringBatch**





## **SpringBatch概述**

​ Spring Batch是个轻量级的、 完善的批处理框架,旨在帮助企业建立健壮、高效的批处理应用。Spring Batch是Spring的一个子项目,使用Java语言并基于Spring框架为基础开发,使得已经使用Spring框架的开发者或者企业更容易访问和利用企业服务。

​ Spring Batch提供了大量可重用的组件,包括了日志追踪、事务、任务作业统计、任务重启、跳过、重复资源管理。对于大数据量和高性能的批处理任  
务,Spring Batch同样提供了高级功能和特性来支持比如分区功能、远程功能。总之,通过Spring Batch能够支持简单的、复杂的和大数据量的批处理作业。

​ Spring Batch是一个批处理应用框架,不是调度框架，但****需要和调度框架（比如美团的：xxl-job）合作来构建完成的批处理任务****。它只关注批处理任务相关的问题,如****事务、并发、监控、执行等****,并不提供相应的调度功能。如果需要使用调用框架，在商业软件和开源软件中已经有很多优秀的企业级调度框架(如Quartz. Tivoli、 Control-M、 Cron等)可以使用。

****框架主要有以下功能:****  
Transaction management (事务管理)  
Chunk based processing (基于块的处理)  
Declarative 1/0 (声明式的输入输出)  
Start/Stop/Restart (启动/停止/再启动)  
Retry/Skip (重试/跳过)

框架一共有****4个主要角色****:

****JobLauncher****是任务启动器，通过它来启动任务，可以看做是程序的入口。

****Job****代表着一个具体的任务。

****Step****代表着一个具体的步骤，一个Job可以包含多个Step (想象把大象放进冰箱这个任务需要多少个步骤你就明白了) .

****JobRepository****是存储数据的地方，****可以看做是一个数据库的接口****，在任务执行的时候需要通过它来记录任务状态等等信息。

## **Job的创建和使用**

****Job****:作业。批处理中的核心概念，是Batch操作的基础单元。  
每个作业Job有1个或者多个作业步Step:

## **Flow的创建和使用**

****1.Flow是多个Step的集合****  
****2.可以被多个Job复用****  
****3.使用FlowBuilder来创建****

## **split实现并发执行**

****实现任务中的多个step或多个flow并发执行****  
1:创建若干个step  
2:创建两个flow  
3:创建-个任务包含以上两个flow,井让这两个flow并发执行

重点：

作业对象 Job

5.1 作业介绍

5.1.1 作业定义

Job作业可以简单理解为一段业务流程的实现，可以根据业务逻辑拆分一个或者多个逻辑块(step)，然后 业务逻辑顺序，逐一执行。 所以作业可以定义为：能从头到尾独立执行的有序的步骤(Step)列表。 有序的步骤列表 一次作业由不同的步骤组成，这些步骤顺序是有意义的，如果不按照顺序执行，会引起逻辑混乱， 比如购物结算，先点结算，再支付，最后物流，如果反过来那就乱套了，作业也是这么一回事。 从头到尾 一次作业步骤固定了，在没有外部交互情况下，会从头到尾执行，前一个步骤做完才会到后一个步 骤执行，不允许随意跳转，但是可以按照一定逻辑跳转。 独立 每一个批处理作业都应该不受外部依赖影响情况下执行。