# 1、下单流程

1、根据商品id与顾客id进行校验

检验本地缓存中有没有该商品id为key的键值在，如果存在说明该商品已售罄，返回，缓存管理工具（google guava），采用LRU淘汰策略

获取redis中的商品库存，如果库存为零则在本地缓存中设置商品id的键值对进行标记

在redis中获取以顾客id与商品id为键值对的值，如果值为1，则该顾客正在秒杀的异步下单中，返回false，否则返回true，------------------------此处的疑问，该处键值对的有效时间为60秒是否合适

2、校验秒杀是否开始或已结束，根据商品秒杀的起止时间

3、redis预减库存，使用decr命令，如果所得结果为小于零，则使用incr命令加回库存，加回库存的原因为保持库存最小为0的状态，以保证有其他订单异常加回库存或者redis与MySQL库存不一致时加回redis库存后可以正常销售，否则预减后不加回，那多次操作后会导致库存为较小的负数，导致业务上库存回流无效

当库存为0是判断redis中有无以promotionId的key存在，该值对应的表内存储库存、秒杀价格，限购数量等信息，如果该key值不存在则通过mq发送一条延时60s的消息，用来进行数据库与redis的同步操作作，然后设置redis中的promotionId，此处没有使用redis的分布式锁，因为如果jvm在发送消息之前挂掉了，那当前产品库存就没有办法在产品销售完之前进行库存同步了，此处可能会多发几条库存同步消息，但是可以容忍

库存同步，当数据库的库存也为0时，则在本地缓存中打上库存售罄标识，如果大于零则重置redis库存，并且删除redis中promotionId标记，为防止重复的重置库存此处引入分布式锁同时加上库存同步版本号(redis一个key,发送消息时加上此版本号)，通过判断promotionId是否存在来判断是否已经同步过库存了

4、异步下单，封装订单信息，封装消息对象，发送给订单队列排队，为实现接口的幂等性，订单中有一个唯一的订单号OrderSn

A、如果消息发送成功，则在redis中添加顾客id商品id的键值对，值为1，表示秒杀排队下单中--------------------------此处过期时间为60S，有疑问，为什么要设置过期时间

B、如果发送消息失败，则首先redis加回库存，然后清理掉本地guava cache中该商品售罄的标志，然后通过redis的发布订阅功能通知集群删除本地缓存该商品的售罄标志

5、回调，如果发送消息出现异常，则进行的回调操作同消息发送失败的操作

# 2、消息

接收到订单下单消息AscynCreateOrderReciever，进行下单操作，mybatis通过selectKey获取订单id，下单成功后redis的顾客与商品id值设为单号值，下单成功后发送延时消息，20分钟内没有支付则删除订单，还原库存，下单成功后如果用户再次请求接口则返回已抢购成功，支付窗口在待支付页面

静态页面存cdn