如何应对缓存穿透和缓存雪崩问题

一般中小型传统软件企业，很难碰到这个问题。如果有大并发的项目，流量有几百万左右。这两个问题一定要深刻考虑。

缓存穿透

即黑客故意去请求缓存中不存在的数据，导致所有的请求都到达数据库，从而数据库连接异常。

解决方案：

1. 利用互斥锁。缓存失效的时候，先去获得锁，得到锁了，再去请求数据库。没得到锁，则休眠一段时间重试。
2. 采用异步更新策略，无论key是否取到值，都直接返回。value值中维护一个缓存失效时间，缓存如果过期，异步起一个线程去读数据库，更新缓存。需要做缓存预热（项目启动前，先加载缓存）操作。
3. 提供一个能迅速判断请求是否有效的拦截机制，比如，利用布隆过滤器，内部维护一系列合法有效的key。迅速判断出请求所携带的key是否合法有效。如果不合法，则直接返回。

缓存雪崩

即缓存同一时间大面积的失效，这个时候又来了一波请求，结果请求都到达数据库，从而导致数据库连接异常。

解决方案：

给缓存的失效时间，加上一个随机值，避免集体失效。

使用互斥锁，但是该方案吞吐量明显下降了。

双缓存。我们有两个缓存，缓存A和缓存B。缓存A的失效时间为20分钟，缓存B不设失效时间。自己做缓存预热操作。然后细分以下几个小点

1. 从缓存A读取数据库，有则直接返回。
2. A没有数据，直接从B读数据，直接返回，并且异步启动一个更新线程。
3. 更新线程同时更新缓存A和缓存B。