

石杉架构-grace同学面经

- 1、自我介绍
- 2、学习经历
- 3、学习中的困惑
- 4、学习方法
- 5、笔记
- 6、学习+面试后的感悟
- 7、面试公司
- 8、面试题
- 9、总结

1、自我介绍

儒猿-石杉架构课

大家好，我是grace，今年32岁，<https://sourl.cn/bFTskZ> 儒猿架构班四期的学员。

毕业于一所普通的本科院校，文科专业。毕业后做电话销售，工资也非常低，难以生存。机缘巧合，我一个高中同学做java开发，工资还不错，为了更好的生存，和我同学交流之后，我也加入了it行业大军，之后进入了教育、旅游等公司做后台开发。

2、学习经历

加入儒猿架构班之后，一直跟着老师的课程，默默的学习，老师给大家规划的学习路线，我就按照老师的节奏来学习。老师的每一课都是大白话，加上经常画图，学起来也非常舒服，易懂。被老师的技术、态度、个人魅力深深的折服，我觉的老师可能是我这辈子能接触到的天花板大牛了。

但是架构班学习的期间因为生娃，带娃，由于家庭原因，家里对带娃的支持给的不是太多，自己这边付出的就要多一点，很艰难，学习过程也是零零散散，断断续续。我觉的这个可能是每个成立家庭的兄弟

后面都要面对的事情，有了家庭，学习的时间就少了，在未成立家庭之前，好好学习，那是多么幸福的事情呀。

但是学习的决心从来没有间断，每次看到老师群里发的大家的励志经历，和师兄弟们市场上千锤百炼，千军万马中拿到一堆offer，都会默默的告诉自己加油。有时候学习的时候精力跟不上，也会打瞌睡，硬着头皮学习。期间也经常看群里大家的交流。

3、学习中的困惑

跟着老师的视频课程学习的时间长了，除了老师自己的笔记，我也经常会对每个专题做笔记总结，但是不知道大家有没有跟我一样的困惑，就是总结的文档一大堆一大堆，时间长了，很长时间就遗忘知识点。这一点，对我的困惑非常大。



4、学习方法

为了解决学习上这个问题，我自己思考，总结了三套方法。

- 1、使用刷题软件，把总结的知识点录制进去，经常没事刷题。
- 2、不断的问自己为什么？为什么要有这个技术？为什么这里这样设计？不这样设计会怎么样。
- 3、总结的时候，要去除繁杂，保留精华，尽量几句话直达本质。先称之为精华学习法吧。

自从使用这个方法后，我就不再弄专题文档了，也是最近才开始，所有问题都精简，大幅度删减，保留核心和精华。几句话能讲清楚的事情，能让别人明白，那就非常牛了，大道至简。

儒猿-石杉架构课

5、笔记

- 1、《从零开始带你成为MySQL实战优化高手》

<https://sourl.cn/bFTskZ>

比如mysql专栏做为案例，架构班的课程姑且不说，就几个专栏的质量实在太强了，原子弹大侠，救火队队长，鞭辟入里的介绍jvm和mysql、rocketmq，这几个专栏是老师朋友开的，学习完，都感觉自己实力有大幅度提升。

不仅老师的朋友强，专栏强，你从学习留言中，还能看到小霸王的身影，哪里都是他。

好，那我就先举几个mysql的例子，这是我学习专栏的时候总结的笔记。

导航		
标题	页面	结果
1) 原理		
2) 异步主从的搭建		
3) 半同步主从		
4) GTID搭建主从		
5) 数据延迟问题		
35. 高可用		
A. 工具介绍		
B. 搭建准备		
C. 搭建步骤		
D. VIP		
36. 案例实战 - 用户表水平拆分		
1) 背景		
2) 解决方案		
37. 案例实战 - 订单数据库设计		
1) 背景		
2) 解决方案		
38. 案例实战 - 跨库分页		
39. 案例实战 - 再次扩容		
二、分库分表		
1. 到底为啥我们要搞麻烦的分库分表?		
2. 分库分表的时候一般是怎么玩儿?		
3. 订单表要根据用户、商家、订单三个维度查, ...		
4. 分布式事务		
5. 查询条件里没包含sharding字段怎么办?		
6. 分库分表的好朋友: 读写分离架构		
7. 分库分表方案设计好之后的物理架构规划		
8. 从单库到分库分表的无缝迁移方案		
9. 针对分库分表的运维管理工作台是什么?		
10. 分库分表的大难题: 扩容, 扩容还是扩容		
11. 通过Sharding-Sphere官网做初步了解		
12. 通过ShardingSphere官网学习功能和原理		
13. 教你如何看ShardingSphere的源码		
14. 订单数据的分库分表方案设计		
15. 落地分库分表方案的几个关键步骤		

第 99 页, 共 126 页 65742 个字 中文(中国)

2) 解决方案

所以说,基本上个这类订单表,哪怕是个小互联网公司,按分库分表几乎是必须得做的,那么怎么做呢?订单表,一般在拆分的时候,往往要考虑到三个维度,一个是必然要按照订单id为粒度去分库分表,也就是把订单id进行hash后,对表数量进行取模然后把订单数据均匀分散到100~1000个表里去,再把这些表分散在多台服务器上。

但是这里有个问题,另外两个维度是用户端和运营端,用户端,就是用户可能要查自己的订单,运营端就是公司可能要查所有订单,那么怎么解决这类问题呢?其实就跟上次的差不多,基本上针对用户端,你需要按照(userid,orderid)这个表结构,去做一个索引映射表。

userid和orderid的一一对应映射关系要放在这个表里,然后针对userid为粒度去进行分库分表,也就是对userid进行hash后取模,然后把数据均匀分散在很多索引映射表里,再把表放在很多数据库里。

然后每次用户端拿出APP查询自己的订单,直接根据userid去hash然后取模路由到一个索引映射表,找到这个用户的orderid,这里当然可以做一个分页了,因为一般订单都是支持分页的,此时可以允许用于用户分页查询orderid,然后拿到一堆orderid了,再根据orderid去按照orderid粒度分库分表的表里提取订单完整数据。

至于运营端,一般都是要根据N多条件对订单进行搜索的,此时跟上次讲的一样,可以把订单数据的搜索条件都同步到ES里,然后用ES来进行复杂搜索,找出来一波orderid,再根据orderid去分库分表里找订单完整数据。

其实大家到最后会发现,分库分表的玩法基本都是这套思路,按业务id分库分表,建立索引映射表同时进行分库分表,数据同步到ES做复杂搜索,基本这套玩法就可以保证你的分库分表场景下,各种业务功能都可以支撑了。

38. 案例实战 - 跨库分页

关于分库分表后的跨库/跨表的分页问题,首先我们先来聊聊这个所谓的分页是个什么场景。那比如说是说之前的那个订单的场景,但用户现在要查询自己的订单,同时订单要求要

儒猿-石杉架构课

<https://sourl.cn/bFTskZ>

通过mysql的学习,解决了我多年的疑问,平时开发过程中,经常搞不清,rr、rc模式下,到底select要不要加手动加锁,几个事务并发执行,这些数据会不会乱,大家之间都有哪些关系。我觉的可能不少人都搞不清楚,写的时候,可能也会糊里糊涂。

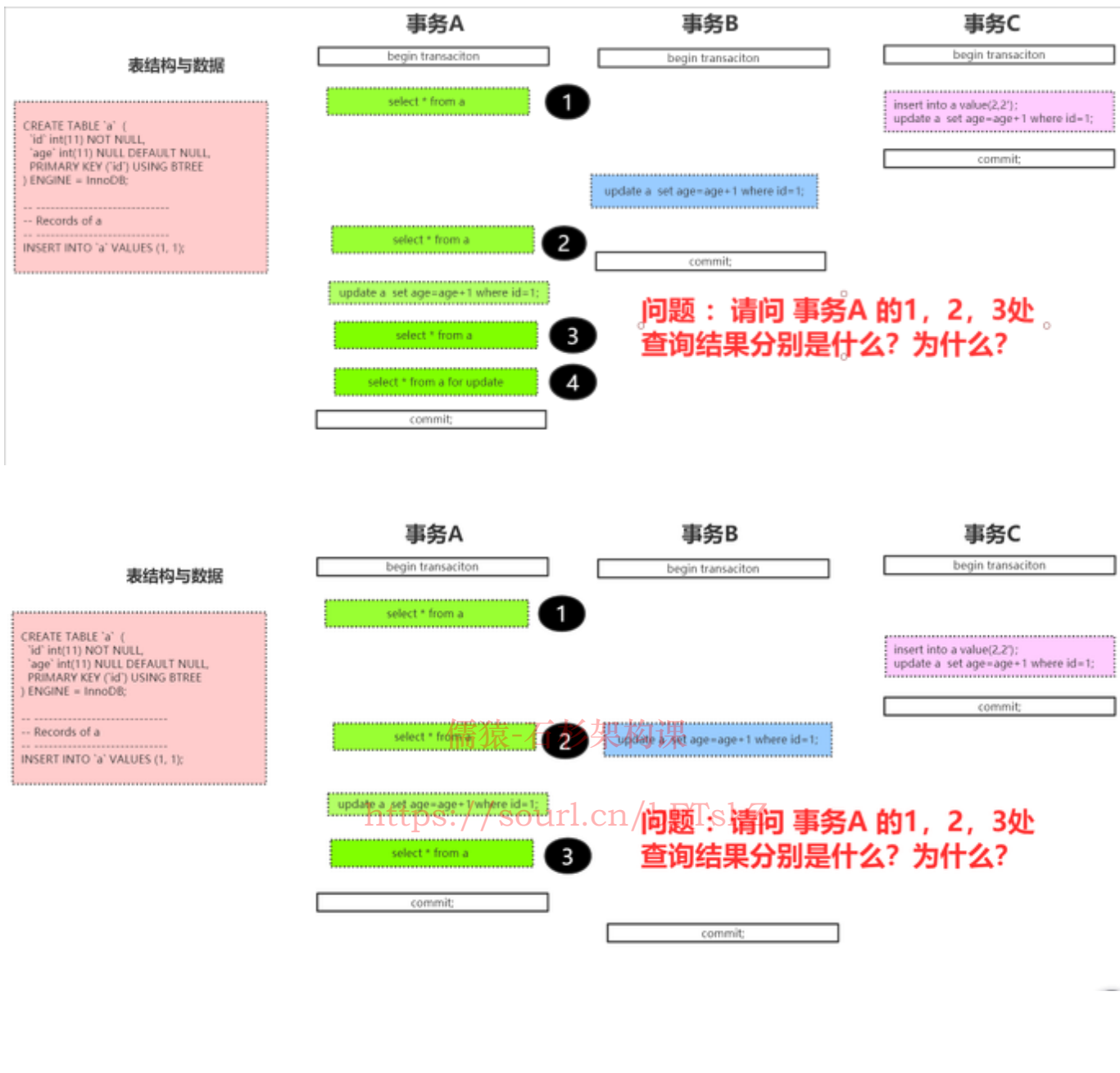
通过msyql专栏的 学习,彻底搞清楚运行原理,写代码的时候,这些种种问题都能甄别。

那么之前为什么学习经历中要分享给同学呢?为什么?

你自己学习知识点,你觉的你消化了,实际上,你不经过一顿毒打和灵魂拷问,很可能只觉的自己掌握了,实际还没有深刻掌握,多几个兄弟拷问你,可能会有你想不到的收获。

就比如群里同学给了两道mysql题,没学过专栏,可能搞不定。

RR模式,看图形红色字给出结果:



2、《从 0 开始带你成为JVM实战高手》

在文档中搜索

标题 页面 结果

16. 案例实战-电商系统

1) 背景

2) 内存使用模型估算

3) 内存分配

4) jvm参数配置

5) 程序执行

6) 新生代垃圾回收优化之一: Survivor空间...

7) 新生代对象经过多少次垃圾回收后进入老...

8) 多大的对象直接进入老年代?

9) 指定垃圾回收器

10) 系统运行模型的估算步骤

11) 老年代的优化

17. G1垃圾回收器

1) parnew+cms的痛点

2) G1设计思路

3) 如何做到垃圾回收时间可控

4) region既可以分配新生代又可以分配老年代

5) 如何设定G1的内存大小

6) Eden和Survivor还存在吗?

7) 控制GC停顿时长

8) 对象什么时间进老年代呢?

9) 大对象怎么办?

10) 什么时候触发新生代和老年代混合垃圾回收

11) G1垃圾回收过程

12) 为什么要多次分批回收?

13) 复制算法回收

14) 哪些region不能回收?

15) 什么时候停止回收

16) 回收失败时的Full GC

18. 案例实战-百万用户教育系统

1) 背景

2) 系统的运行压力

3) G1内存布局

4) GC停顿时间如何设置?

7) 控制 GC 停顿时长

-XX:MaxGCPauseMills, 默认值是 200ms

8) 对象什么时间进老年代呢?

达到 15 岁 -XX:MaxTenuringThreshold

动态年龄判断, 年龄 n 之前的超过 50%容量了, n 之后的进老年代

9) 大对象怎么办?

如果对象大小超过 region 的一半, 就要进大对象 region。

空的 region 不属于新生代, 也不属于老年代, 大对象就可以使用。

对象太大的话, 可以横跨多个 region

新生代、老年代回收的时候, 会顺带着回收大对象

10) 什么时候触发新生代和老年代混合垃圾回收

-XX:InitiatingHeapOccupancyPercent, 默认值是 45%

老年代占据了总 region 的 45%, 就触发混合垃圾回收

11) G1 垃圾回收过程

初始标记: 根据 GCRoots 是否可达, 标记对象是否是垃圾, stw, 速度很快

儒猿-石杉架构课

30 页, 共 104 页 39413 个字 中文(中国)

<https://sourl.cn/bFTskZ>

3、并发举例

架构课里的并发讲解，包括并发源码和实战，老师一套并发讲下来，自己再进行思考和总结。整理了200多页的学习笔记

6

导航

在文档中搜索

标题

页面

结果

4) 写一个固定容量的同步容器

5) 写一个生产者消费者

12. 实战案例-注册中心

1) 注册中心背景

2) 客户端 注册心跳

3) 服务端 注册心跳 故障检测

4) Join 改造客户端 注册心跳

5) Interrupt 优雅关闭心跳线程

6) volatile优化 优雅关闭机制

7) volatile优化 存活状态检查机制

8) 客户端定时拉取注册表

9) 加锁保证服务端注册表读写线程安全

10) 服务下线

11) 自我保护

1) Atomic优化心跳计数器

2) 拉取全量和增量注册表

3) AtomicReference优化客户端缓存注册表

4) AtomicStampedReference优化注册表

5) AtomicLong确保多线程拉取注册表版本...

6) 读写锁 优化注册表读写

7) 三级缓存

8) Map改为ConcurrentHashMap

9) 最近改变队列改为ConcurrentLinkedQue...

10) 三层队列同步机制

11) 线程池处理集群同步任务

13. 实战案例-分布式文件存储系统

1) 背景

2) edits log机制介绍

3) NameNode代码基本框架实现

4) 创建目录功能实现

5) edits log的全局xid机制以及双缓冲机制...

6) synchronized分段加锁机制和wait与notif...

7) Slave节点向多个Master注册

8) CopyOnWriteArrayList保存ServiceActor

2) 拉取全量和增量注册表

a. 服务注册中心的最近更新服务实例队列实现

每隔 30 秒就拉取一次全量的注册表呢？就是说这个注册表可能并不是每一条数据都会变化的，每隔 30 秒，可能就少数几个服务实例会出现这个变化，并不需要每隔 30 秒吭吭吭把所有的注册表的数据都拉取过来

如果说你的服务实例有几万个，那么你的服务注册中心的注册表里可能有几万条数据，每次你得 30 秒拉取几万条数据的服务注册表，很坑爹的，对整个网络的开销，每次对服务注册中心的性能压力，都很大

服务启动的时候，先是全量拉取一次服务注册表，接着每隔 30 秒，拉取一次增量的注册表，这样的话，每隔 30 秒就是拉取最近 30 秒变化的少量的服务实例的信息即可

增量拉取注册表如何实现呢？你搞一个队列，队列里面就是存放最近 3 分钟有变化的服务实例

上线、下线的时候，把实例信息放入三分钟改变队列。

启动一个定时任务，定期把超过 3 分钟的队列数据移除出队列。

```
package com.happy.server;

import lombok.*;

import java.util.HashMap;
import java.util.LinkedList;
import java.util.Map;
```

第 143 页, 共 205 页 48264 个字 中文(中国)

儒猿-石杉架构课

<https://sourl.cn/bFTskZ>

6、学习+面试后的感悟

1、化繁为简，记忆本质，减少大脑负担

面试的时候，面试官没有时间听你一堆唠叨，也没有时间听你说的很细很细的，会经常打断你，我的实际情况是这样的。每个面试官的风格也不一样。站在我自己的角度，我希望，以最短的时间震撼面试官。

如果能画图，也非常nice；不过我不喜欢画图，我个人感觉现场画图，可能短时间讲的东西太少。我更喜欢口述，以精简的语言，达到最好的效果。大家看自己的情况。找到适合自己的方式。

2、脚踏实地，静心治学

学习是一个长期的过程，不急不躁，慢工出细活。

可以多和儒猿群友交流学习成果，每个人对专栏和架构课程的领悟都不一样。

3、核心能力

越往上走，很多公司，要求你有带团队的能力，给团队成员提供指导和帮助，技术和管理双要求，说白了就是管理好人；技术上，要求你有架构设计的能力，业务建模 + 技术架构，说白了，就是能把项目做的好。

个人技术栈按照老师给的学习路线，持续打造，已经超级强了。关键在于自己的消化和吸收，领悟，内化。形成自己的知识，学以致用，融合一体。

这里老师的课程高并发、高可用、海量数据、中间件源码，几十门课，不得不感叹老师的博大精深，好多门课一起更新，听老师课时间久的都知道老师嗓子经常不舒服，天天讲课，肯定有影响的，希望老师能保护下嗓子。

儒猿-石杉架构课

4、使用平板

平板可以倍速，可以拖拽，可以带着到处跑，我感觉使用了平板之后，我的学习效率大幅度提升，很舒服。

7、面试公司

1、这次面试没有特别准备

就是加入架构班以来的静心学习，师兄弟之间的交流，老师的问询关怀，自己遇到问题尝试自己去解决，解决不了的寻求老师帮助。

简历准备，从师兄那里要了两份模板，简历内容已经非常强了，按照那个改了改。

2、面试公司介绍和经过

某线下零售商超：

这块全程问我项目，就是架构师云实战seckill项目。当初因为我自己没有通过云实战 大规模服务器 跑一边流程，脑补了一些东西，对项目的准备也不足，所以有些点答的不是太好。

事后跟老师沟通，老师要求严格按照实战项目的数据指标来。同时需要对自己树立信心。很多面试官技术都不如咱们。经过老师指导，稍作整理，再战3家公司，这次带着自信，面试基本上气场、自信、被认可的感觉都非常ok

这个是面试结束后，找老师复盘的部分聊天：

```
grace 2021/5/15 星期六 21:41:02
    第4点，本来是mq消息，10分钟超时丢弃还库存，带出来的mq积压消息处理方案。
石杉 2021/5/15 星期六 21:49:54
    1.库存分片，面试官说，不用集群，宕机了，那台机器硬件损坏拉不起来，带来了销售量的损失。跟我说用集群，不会有库存的丢失，我说异步复制会丢数据，人家貌似不太认可我这种单机自己管理分片的方式。

    他不认可，说明他自己太菜，根本不懂这个问题，如果用集群异步复制，宕机就会导致库存超卖，这是必然的
石杉 2021/5/15 星期六 21:50:24
    2.你们redis宕机失败了，有其他方案保证吗，我说没有，这里我做了强一致，抛弃了可用性。对方提示，你们不用jvm缓存吗，我说用了jvm缓存，就保证不了库存的一致性，貌似对方也不太满意。

    他们太菜，redis如果故障了，他可能想要说是可以直接和mysql，但是实际上根本不可能
石杉 2021/5/15 星期六 21:50:54
    3.发redis失败，发mq失败，降级写内存写磁盘，人家说写内存jvm内存打满oom，写磁盘太慢，我说nio顺序写不慢。对方貌似不太认可我这种做法。问我观测过降级后的线上数据没。我说压根没问题。都写入磁盘了。

    他们实在是太菜太菜了，顺序写磁盘，kafka都是这么搞的，难道kafka不行？
石杉 2021/5/15 星期六 21:51:29
    4.mq消息积压，问怎么处理，我说扩机器和丢消息，人家说不允许扩机器，不丢消息，自己设计线程池消费消息，我说采用fix线程池自己写queue队列，queue数量超过的，让mq稍后再推。

    面试官不认可，说自定义线程池然后采用那个拒绝策略【使用调用线程来处理】也就是使用tomcat的线程来处理。

    然后问线程池的线程数量应该怎么设置，我说根据任务耗时，来计算。可能这里回答的不太好，面试官的意思大概是你既然说了秒杀250个线程最佳，这里下单也要复用这250个线程的意思。

    面试官拿到一定程度了
    如果消息积压，一定是订单系统处理太慢了，或者你的秒杀量太大太大了
    一般就是直接降级降级，把秒杀库存还回去，全部让秒杀失败，failfast思想
    缓解压力，保证一部分秒杀请求正常执行
    失败的人可以再次秒杀
```

<https://sourl.cn/bFTskZ>

传统行业龙头公司

一面 面试官 给的问题，基本上，我按照我的思路，精简+本质，一两句话概括，没几个问题，面试官就不问了，觉的技术能力很强了。然后就开始出场景题，我也按照我的日常经验给了一些解决方案。然后hrbp聊了很长时间，就约了二面。

二面是一个很高级的负责人了，我们聊了30分钟，全程就是我自己自我介绍，讲述了一个云实战项目，介绍一下自己是怎么样的人，就结束了，第二天给了60w的总包。考虑的离家近不加班，再好好沉淀跟老师学习，最终决定去这家公司。

行业龙头，约了后，写笔试题，普通的笔试题，我基本上概括语言写出不一样的高度，技术面就过了。二面领导就聊项目设计的经验。hr电话谈薪资给了65w左右的包，我提了70w，后面hr跟领导争取，没有通过

hr的意思也是公司领导希望用大量校招来补充。加上这边公司大小周，距离有点远，有娃了，也不能跑那么远，跑远了一加班，晚上就回不了家了，因此我也拒绝了这个offer。

7、推荐营销

小公司，还是一样的。一面 二面 两个面试官轮流过，技术上都ok了，三面hr跟我聊了一个小时，要了70w，对方也能给到。

后面脉脉上搜了下，公司口碑太差了，我面试过程中也能感受到几个面试官一边打击一边想拉你入伙，加上离家也远，加班强度高，就没接了。儒猿-石杉架构课

最终，第三家公司，各方面都比较适合我，选择第三家公司，offer已接，准备在本月入职。

8、面试题

我出去面试，感觉面试官问我的技术题不会多，可能是我都每个点的回答都能快速、简短、直达本质，加上项目也很强，直接震慑了面试官。

举个例子来说，比如jmm，我一套打下来，面试官可能还没等我说完就打断我，直接下一个。分布式redis锁，我把我自己深入去研究的结果说一遍，可能还捎带一些细节，比如它客户端使用了semaphore + pub sub，就感觉已经很强了，面试官一般这个话题就会终结。

相信大家市场上能接触到的面试题已经很多很多了，不需要我再提供了，老师也给了很多跳槽指导，题目定制，也结合每个人的自身进行项目定制，简历定制。

我的感悟就是，脚踏实地的技术栈积累，学而不思则罔，跟着老师学习渔的能力，多思考本质性内容，多问为什么为什么为什么！！！！

附录：面试题

1、技术问题

1. ip隧道原理
 2. cdn缓存原理
 3. http和https的区别
 4. http报文内容介绍一下
 5. https原理
 6. 谈谈你对安全防护各个方面的理解
 7. 为什么零拷贝更快，具体原理
 8. mysql事务acid是如何实现的
 9. mysql幻读是如何解决的
 10. mysql rr模式的原理
 11. redis锁和zk锁的实现原理
- 儒猿-石杉架构课
<https://soul.cn/bFTskZ>
-

12. 谈谈你对锁优化的理解
 13. 谈谈你对JMM的理解
 14. 谈谈你对AQS的理解
 15. 谈谈你对读写锁的理解
 16. 谈谈有哪些保证原子性的手段
 17. 说一说你对ddd 战术部分的理解，代码如何分层
 18. 聊聊spring cloud各个组件的原理
 19. hystrix限流是怎么做的，原理
 20. 谈一谈你对CompletableFuture的理解，为什么要用这个，超时机制怎么实现的
 21. rocketmq中间件里面有哪些组件，说一说都有什么作用
 22. mq事务消息的实现原理
 23. 如何解决缓存热key问题
-

2、设计题

1、你平时是如何设计项目的？

使用uml进行业务流程的建模，映射系统用例，对系统用例进行领域分析出类图。
对系统用例，进行功能的分析序列图设计，完整阐释领域类之间如何协作完成功能。
然后是db的设计，接口契约的设计，工时的预估。

2、设计一个类似于微博消息系统，能提示消息未读，对已读未读做统计，不要用redis，原始数据结构的设计。

当时我给了几张表，发布主题的消息表，个人维度的消息读取数量的统计表，已读消息序号增长表。

3、设计一个千万级别的分库分表，多维度如何处理？复杂查询如何处理？扩容如何处理？

当时我的回答是，用shardingjdbc分库分表，多维度也按不同的维度字段分。复杂查询数据canal监听binlog导入es

(1) 幂等你们是怎么做的

(2) 说一说你们现在分布式id的实现方案，有什么缺点

3、开放题

儒猿-石杉架构课

1、你觉的你是一个什么样的人？

自我驱动，勇于担当，主动沟通和推进项目，能和团队成员良好协作，完成部门内和跨部门协作任务。

2、你觉的你能胜任我们提供的这个职位吗？

完全可以，工作态度认真积极，自我驱动力强，勇于挑战。

9、总结

感谢石杉老师，感谢各位专栏老师，感谢各位助教和工作人员。谢谢你们！

千言万语，只能用实力和结果，来证明自己和报答各位老师
