

你知道的越多，你不知道的越多

点赞再看，养成习惯

本文 **GitHub** <https://github.com/JavaFamily> 上已经收录，有一线大厂面试题思维导图，也整理了很多我的文档，欢迎Star和完善，大家面试可以参照考点复习，希望我们一起有点东西。

前言

作为一个在互联网公司面一次拿一次Offer的面霸，打败了无数竞争对手，每次都只能看到无数落寞的身影失望的离开，略感愧疚（请允许我使用一下夸张的修辞手法）。

于是在一个寂寞难耐的夜晚，我痛定思痛，决定开始写互联网技术栈面试相关的文章，希望能帮助各位读者以后面试势如破竹，对面试官进行360°的反击，吊打问你的面试官，让一同面试的同僚瞠目结舌，疯狂收割大厂Offer！

所有文章的名字只是我的噱头，我们应该有一颗谦逊的心，所以希望大家怀着空杯心态好好学，一起进步。

面试开始

一个大腹便便，穿着格子衬衣的中年男子，拿着一个满是划痕的mac向你走来，看着快秃顶的头发，心想着肯定是尼玛顶级架构师吧！但是我们腹有诗书气自华，虚都不虚。



小伙子您好，看你简历上写了你项目里面用到了Redis，你们为啥用Redis？

心里忍不住暗骂，这叫啥问题，大家不都是用的这个嘛，但是你不能说出来。

认真回答道：帅气迷人的面试官您好，因为传统的关系型数据库如Mysql已经不能适用所有的场景了，比如秒杀的库存扣减，APP首页的访问流量高峰等等，都很容易把数据库打崩，所以引入了缓存中间件，目前市面上比较常用的缓存中间件有Redis 和 Memcached 不过中和考虑了他们的优缺点，最后选择了Redis。

至于更细节的对比朋友们记得查阅Redis 和 Memcached 的区别，比如两者的优缺点对比和各自的场景，后续我有时间也会写出来。

那小伙子，我再问你，Redis有哪些数据结构呀？

字符串String、字典Hash、列表List、集合Set、有序集合SortedSet。

这里我相信99%的读者都能回答上来Redis的5个基本数据类型。如果回答不出来的小伙伴我们就要加油补课哟，大家知道五种类型最适合的场景更好。

但是，如果你是Redis中高级用户，而且你要在这次面试中突出你和其他候选人的不同，还需要加上下面几种数据结构HyperLogLog、Geo、Pub/Sub。

如果你还想加分，那你说还玩过Redis Module，像BloomFilter，RedisSearch，Redis-ML，这个时候面试官得眼睛就开始发亮了，心想这个小伙子有点东西啊。

注：本人在面试回答到Redis相关的问题的时候，经常提到BloomFilter（布隆过滤器）这玩意的使用场景是真的多，而且用起来是真的香，原理也好理解，看一下文章就可以在面试官面前侃侃而谈了，不香么？

如果有大量的key需要设置同一时间过期，一般需要注意什么？

如果大量的key过期时间设置的过于集中，到过期的那个时间点，redis可能会出现短暂的卡顿现象。严重的话会出现缓存雪崩，我们一般需要在时间上加一个随机值，使得过期时间分散一些。

电商首页经常会使用定时任务刷新缓存，可能大量的数据失效时间都十分集中，如果失效时间一样，又刚好在失效的时间点大量用户涌入，就有可能造成缓存雪崩

那你使用过Redis分布式锁么，它是怎么回事？

先拿setnx来争抢锁，抢到之后，再用expire给锁加一个过期时间防止锁忘记了释放。

这时候对方会告诉你说你回答得不错，然后接着问如果在setnx之后执行expire之前进程意外crash或者要重启维护了，那会怎么样？

这时候你要给予惊讶的反馈：唉，是喔，这个锁就永远得不到释放了。紧接着你需要抓一抓自己得脑袋，故作思考片刻，好像接下来的结果是你主动思考出来的，然后回答：我记得set指令有非常复杂的参数，这个应该是可以同时把setnx和expire合成一条指令来用的！

对方这时会显露笑容，心里开始默念：嗯，这小子还不错，开始有点意思了。

假如Redis里面有1亿个key，其中有10w个key是以某个固定的已知的前缀开头的，如何将它们全部找出来？

使用keys指令可以扫出指定模式的key列表。

对方接着追问：如果这个redis正在给线上的业务提供服务，那使用keys指令会有什么问题？

这个时候你要回答redis关键的一个特性：redis的单线程的。keys指令会导致线程阻塞一段时间，线上服务会停顿，直到指令执行完毕，服务才能恢复。这个时候可以使用scan指令，scan指令可以无阻塞的提取出指定模式的key列表，但是会有一定的重复概率，在客户端做一次去重就可以了，但是整体所花费的时间会比直接用keys指令长。

不过，增量式迭代命令也不是没有缺点的：举个例子，使用SMEMBERS命令可以返回集合键当前包含的所有元素，但是对于SCAN这类增量式迭代命令来说，因为在对键进行增量式迭代的过程中，键可能会被修改，所以增量式迭代命令只能对被返回的元素提供有限的保证。

使用过Redis做异步队列么，你是怎么用的？

一般使用list结构作为队列，rpush生产消息，lpop消费消息。当lpop没有消息的时候，要适当sleep一会再重试。

如果对方追问可不可以不用sleep呢？

list还有个指令叫**blpop**，在没有消息的时候，它会阻塞住直到消息到来。

如果对方接着追问能不能生产一次消费多次呢？

使用pub/sub主题订阅者模式，可以实现 1:N 的消息队列。

如果对方继续追问 pub/sub有什么缺点？

在消费者下线的情况下，生产的消息会丢失，得使用专业的消息队列如**RocketMQ**等。

如果对方究极TM追问Redis如何实现延时队列？

这一套连招下来，我估计现在你很想把面试官一棒打死（面试官自己都想打死自己了怎么问了这么多自己都不知道的），如果你手上有一根棒球棍的话，但是你很克制。平复一下激动的内心，然后神态自若的回答道：使用sortedset，拿时间戳作为score，消息内容作为key调用zadd来生产消息，消费者用**zrangebyscore**指令获取N秒之前的数据轮询进行处理。

到这里，面试官暗地里已经对你竖起了大拇指。并且已经默默给了你A+，但是他不知道的是此刻你却竖起了中指，在椅子背后。

Redis是怎么持久化的？服务主从数据怎么交互的？

RDB做镜像全量持久化，AOF做增量持久化。因为RDB会耗费较长时间，不够实时，在停机的时候会导致大量丢失数据，所以需要AOF来配合使用。在redis实例重启时，会使用RDB持久化文件重新构建内存，再使用AOF重放近期的操作指令来实现完整恢复重启之前的状态。

这里很好理解，把RDB理解为一整个表全量的数据，AOF理解为每次操作的日志就好了，服务器重启的时候先把表的数据全部搞进去，但是他可能不完整，你再回放一下日志，数据不就完整了嘛。不过Redis本身的机制是 AOF持久化开启且存在AOF文件时，优先加载AOF文件；AOF关闭或者AOF文件不存在时，加载RDB文件；加载AOF/RDB文件成功后，Redis启动成功；AOF/RDB文件存在错误时，Redis启动失败并打印错误信息

对方追问那如果突然机器掉电会怎样？

取决于AOF日志sync属性的配置，如果不要求性能，在每条写指令时都sync一下磁盘，就不会丢失数据。但是在高性能的要求下每次都sync是不现实的，一般都使用定时sync，比如1s1次，这个时候最多就会丢失1s的数据。

对方追问RDB的原理是什么？

你给出两个词汇就可以了，fork和cow。fork是指redis通过创建子进程来进行RDB操作，cow指的是copy on write，子进程创建后，父子进程共享数据段，父进程继续提供读写服务，写脏的页面数据会逐渐和子进程分离开来。

注：回答这个问题的时候，如果你还能说出AOF和RDB的优缺点，我觉得我是面试官在这个问题上我会给你点赞，两者其实区别还是很大的，而且涉及到Redis集群的数据同步问题等等。想了解的伙伴也可以留言，我会专门写一篇来介绍的。

Pipeline有什么好处，为什么要用pipeline?

可以将多次IO往返的时间缩减为一次，前提是pipeline执行的指令之间没有因果相关性。使用redis-benchmark进行压测的时候可以发现影响redis的QPS峰值的一个重要因素是pipeline批次指令的数目。

Redis的同步机制了解么?

Redis可以使用主从同步，从从同步。第一次同步时，主节点做一次bgsave，并同时后续修改操作记录到内存buffer，待完成后将RDB文件全量同步到复制节点，复制节点接受完成后将RDB镜像加载到内存。加载完成后，再通知主节点将期间修改的操作记录同步到复制节点进行重放就完成了同步过程。后续的增量数据通过AOF日志同步即可，有点类似数据库的binlog。

是否使用过Redis集群，集群的高可用怎么保证，集群的原理是什么?

Redis Sentinel着眼于高可用，在master宕机时会自动将slave提升为master，继续提供服务。

Redis Cluster着眼于扩展性，在单个redis内存不足时，使用Cluster进行分片存储。

面试结束

小伙子你可以的，什么时候有时间来上班啊，要不明天就来吧?

你强装镇定，这么急啊我还需要租房，要不下礼拜一吧。

好的 心想这小子这么NB是不是很多Offer在手上，不行我得叫hr给他加钱。

能撑到最后，你自己都忍不住自己给自己点个赞了!

(暗示点赞，每次都看了不点赞，你们想白嫖我么? 你们好坏哟，不过我喜欢)。

总结

在技术面试的时候，不管是Redis还是什么问题，如果你能举出实际的例子，或者是直接说自己开发过的问题和收获会给面试官的印象分会加很多，回答逻辑性也要强一点，不要东一点西一点，容易把自己都绕晕的。

还有一点就是我问你为啥用Redis你不要一上来就直接回答问题了，你可以这样回答：

帅气的面试官您好，首先我们的项目DB遇到了瓶颈，特别是秒杀和热点数据这样的场景DB基本上就扛不住了，那就需要缓存中间件的加入了，目前市面上有的缓存中间件有 Redis 和 Memcached，他们的优缺点.....，综合这些然后再结合我们项目特点，最后我们在技术选型的时候选了谁。

如果你这样有条不紊，有理有据的回答了我的问题而且还说出这么多我问题外的知识点，我会觉得你只是一个会写代码的人，你逻辑清晰，你对技术选型，对中间件对项目都有自己的理解和思考，说白了就是你的offer有戏了。

点关注，不迷路

好了各位，以上就是这篇文章的全部内容了，能看到这里的人呀，都是人才。

我后面会每周都更新几篇一线互联网大厂面试和常用技术栈相关的文章，非常感谢人才们能看到这里，如果这个文章写得还不错，觉得「敖丙」我有点东西的话 求点赞👍 求关注❤️ 求分享👥 对暖男我来说真的 非常有用!!!

创作不易，各位的支持和认可，就是我创作的最大动力，我们下篇文章见！

敖丙 | 文 【原创】

如果本篇博客有任何错误，请批评指教，不胜感激！

文章每周持续更新，微信搜索「三太子敖丙」第一时间阅读和催更（比博客早一到两篇哟），本文 **GitHub** <https://github.com/JavaFamily> 上已经收录，有一线大厂面试点思维导图，也整理了很多我的文档，欢迎Star和完善，大家面试可以参照考点复习，希望我们一起有点东西。