\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第0篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

一次Java社招面试经历  
  
编辑于 2020-08-23 09:04:59  
  
 1、HashMap是不是有序的？（不是有序的）   
 2、有没有有顺序的Map实现类？（有TreeMap和LinkedHashMap）   
 3、TreeMap和LinkedHashMap是如何保证它的顺序的？你觉得它们两个哪个的有序实现比较好？你觉得还有没有比它更好或者更高效的实现方式？   
 4、如果想实现所有的线程一起等待某个事件的发生，当某个事件发生时，所有线程一起开始往下执行的话，有什么好的办法吗？   
 答：可以用栅栏（Java的并发包中的CyclicBarrier）   
   
 5、那你知道它的实现原理吗？你还知道其它的实现方式吗？你觉得这些方式里哪个方式更好？如果让你来写的话，你觉得还有比它更好的实现方式吗？   
 6、NIO模型，特别是其中的selector的职责和实现原理。   
 7、NIO的核心是什么？（IO线程池）   
 8、问IO包的设计模式（装饰器模式），为什么要这样设计？有没有更好的设计？   
 9、什么时候一个对象会被GC？   
 10、接着问为什么要在这种时候对象才会被GC？   
 11、GC策略都有哪些分类？这些策略分别都有什么优劣势？都适用于什么场景？   
 12、举个实际的场景，让选择一个GC策略？然后问什么要选择这个策略？   
 13、Java的类加载器都有哪些？每个类加载器都加载哪些类？这些类加载之间的父子关系是怎样的？   
 14、什么是双亲委派模型？为什么Java的类加载器要使用双亲委派模型？   
 15、如何自定义自己的类加载器，自己的类加载器和Java自带的类加载器关系如何处理？   
 16、内存分为哪几部分，这些部分分别都存储哪些数据？   
 17、一个对象从创建到销毁都是怎么在这些部分里存活和转移的？   
 18、内存的哪些部分会参与GC的回收？   
 19、Java的内存模型是怎么设计的？为什么要这么设计？   
 20、请结合内存模型的设计谈谈volatile关键字的作用？   
 21、讲做过的项目，然后会让你去解释其中某一部分，比如让你解释当时为什么要这么做，或者问你现在觉得有没有更好的办法，所用到的技术哪些？

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第1篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴社招Java工程师面经  
  
发布于 2020-08-21 09:34:30  
  
 1、ArrayList、LinkedList、Vector的区别   
 （1）baiArrayList、Vector、LinkedList类都是java.util包中，均为可伸缩数组。   
 （2）ArrayList和Vector底层都是数组实现的，所以索引数据快，删除、插入数据慢。   
 （3）ArrayList采用异步的方式，性能好，属于非线程安全的操作类。Vector采用同步的方式，性能较低，属于线程安全的操作类。   
 （4）LinkedList底层是链表实现，所以，索引慢，删除、插入快，属于非线程安全的操作类。   
     
 2、HashMap和HashTable的区别，Hashmap实现原理   
 3、String和StringBuffer的区别   
 （1）String:   
      •是对象不是原始类型。   
      •为不可变对象,一旦被创建,就不能修改它的值。   
      •对于已经存在的String对象的修改都是重新创建一个新的对象,然后把新的值保存进去。   
      •String是final类,即不能被继承。   
     
 （2）StringBuffer:   
     •是一个可变对象,当对它进行修改的时候不会像String那样重新建立对象。   
     •它只能通过构造函数来建立，StringBuffer subffer=new StringBuffer()；   
     •对象被建立以后,在内存中就会分配内存空间,并初始保存一个null,通过它的•append方法向其赋值 subffer.append(“hello word”)；   
     
 （3）字符串连接操作中StringBuffer的效率要明显比String高;   
 String对象是不可变对象,每次操作String都会建立新的对象来保存新的值。   
 StringBuffer对象实例化后,只对这一个对象操作。   
     
     
 4、gc的概念，如果A和B对象循环引用，是否可以被GC？   
 5、Java中的内存溢出是如何造成的 ，知道内存泄漏吗？原因？   
 6、String s = “123”;这个语句有几个对象产生   
 7、Exception和RuntimeException的区别，作用又是什么？   
 8、java异常体系知道吗？error和exception有什么区别？   
 9、FileInputStream与FileReader区别。   
 FileInputStream是字节流，FileReader是字符流，用字节流读取中文的时候，可能会出现乱码，而用字符流则不会出现乱码，而且用字符流读取的速度比字节流要快；   
     
 10、char型变量中能不能存贮一个中文汉字?为什么?   
 11、List、Set、Map是否都继承自Collection接口？ 他们的区别   
 12、java反射中的应用场景   
 13、注解用过吗？作用是什么？   
 14、java编译时和运行时有什么区别？   
 15、知道socket吗？和Websocket有什么区别？

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第2篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里后端Java社招面经  
  
编辑于 2020-08-20 17:04:43  
  
 一面：   
 1、谈谈你对HashMap的理解，底层原理的基本实现，HashMap是怎么解决碰撞问题的？   
 2、开发中用了那些数据库？回答mysql，储存引擎有哪些？然后问了我悲观锁和乐观锁问题使用场景、分布式集群实现的原理。   
 3、springmvc和mybatis的工作原理，有没有看过底层源码？   
 4、熟悉IO么？与NIO的区别，阻塞与非阻塞的区别。   
 5、海量数据分析。   
 6、线程安全和非线程安全。   
 7、HTTP2.0、thrift。   
 8、java反射应用。   
 9、分布式事务一致性。   
 10、nio的底层实现。   
 11、jvm基础是必问的，jvm GC原理，JVM怎么回收内存。   
 12、API接口与SDI接口的区别。   
 13、dubbo如何一条链接并发多个调用。Dubbo的原理，序列化相关问题。   
 14、用过哪些中间件？   
 15、做过工作流引擎没有？   
 16、以前的工作经历，自己觉得有哪些出彩的地方？   
 17、线程池的一些原理，锁的机制升降级。   
 18、从系统层面考虑，分布式从哪些纬度考虑？   
 19、Hadoop底层怎么实现。   
 20、threadLocal，线程池，hashMap/hashTable/coccurentHashMap等。   
 21、秒杀系统的设计。   
 22、虚拟机，IO相关知识点。   
 23、Linux的命令。   
 24、一个整形数组，给定一个数，在数组中找出两个数的和等于这个数，并打印出来，我写的时间复杂度高，要求O(n)。   
 25、n个整数，找出连续的m个数加和是最大。   
   
 二面：   
 1、数据库锁隐原理。   
 2、1000个线程同时运行，怎么防止不卡？   
 3、并列的并发消费问题。   
 4、高并发量大的话怎么处理热点，数据等。   
 5、如何获取一个本地服务器上可用的端口？   
 6、流量控制相关问题。   
 7、数据库TPS是多少，是否进行测试过？   
 8、缓存击穿有哪些方案解决？   
 9、Java怎么挖取回收器相关原理？   
 10、Java的集合都有哪些，都有什么特点？   
 11、分布式锁，redis缓存，spring aop，系统架构图，MySQL的特性。   
 12、场景，同时给10万个人发工资，怎么样设计并发方案，能确保在1分钟内全部发完。打个比方会提出类似的场景。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第3篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴Java面试经历分享  
  
编辑于 2020-08-20 16:41:41  
  
 1、String和StringBuffer的区别。   
 2、gc的概念，如果A和B对象循环引用，是否可以被GC？   
 3、Java中的内存溢出是如何造成的。   
 4、String s = “123”；这个语句有几个对象产生。   
 5、Error、Exception和RuntimeException的区别，作用又是什么？   
 6、列举3个以上的RuntimeException。   
 7、reader和inputstream区别。   
 8、hashCode的作用。   
 9、Object类中有哪些方法，列举3个以上（可以引导）   
 10、char型变量中能不能存贮一个中文汉字?为什么？   
 11、列举几个Java Collection类库中的常用类。   
 12、List、Set、Map是否都继承自Collection接口？   
 13、HashMap和Hashtable的区别。   
 14、HashMap中是否任何对象都可以做为key,用户自定义对象做为key有没有什么要求？   
 15、interface 和 abstrat class的区别 是否可以继承多个接口，是否可以继承多个抽象类。   
 16、 启动一个线程是用run()还是start()？   
 17、多线程有几种实现？同步和并发是如何解决的？   
 18、什么叫守护线程？用什么方法实现守护线程（Thread.setDeamon()的含义）   
 19、如何停止一个线程？ 解释是一下什么是线程安全？举例说明一个线程不安全的例子。   
 20、解释Synchronized关键字的作用。   
 21、当一个线程进入一个对象的一个synchronized方法后，其它线程是否可进入此对象的其它方法？   
 22、了解过哪些JDK8的新特性，举例描述下相应的特性？   
 23、对sql进行优化的原则有哪些？   
 24、servlet生命周期是生命与cgi的区别？   
 25、StringBuffer有什么优势？为什么快？   
 （1）stringbuffer和stringbuilder一般用来和string来比较，一般情况下的string，只要发生改变，就会新生成一个对象，如果改变的多了的话，比如一些科学计算，计算次数上千万，会很消耗内存。这个时候你就可以用stringbuffer和stringbuilder来代替，它们在发生改变的时候，不会像string一样不停地新生成对象，所以可以节省内存。   
 （2）而stringbuffer和stringbuilder之间的区别在于一个是线程安全，一个是线程不安全。线程安全的，在进行并发处理的时候，会有阻塞现象，而线程不安全的在遇到并发的时候就不会阻塞，所以线程安全的会比线程不安全的慢。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第4篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里Java岗社招面试经历分享  
  
编辑于 2020-08-14 10:08:02  
  
 笔试   
   
 耗时三个小时，考的是Java核心的基础。大概说一下几个考点：   
 Java并发的知识点、集合类、线程池、多线程之间的通信等   
   
 一面   
   
 1、首先是自我介绍。个人情况，个人的擅长的地方，这里主要考察你的表达能力和语言精简能力。   
 2、介绍自己的项目经验，项目中做了什么，难点在哪里？主要是想了解参与过技术难度最大的项目难点，技术难点在哪里，你是怎么来解决的？   
 3、Java的线程池说一下，各个参数的作用，如何进行的？   
 4、同步与异步区别？   
 （1）同步，可以理解为在执行完一个函数或方法之后，一直等待系统返回值或消息，这时程序是出于阻塞的，只有接收到返回的值或消息后才往下执行其他的命令。   
 （2）异步，执行完函数或方法后，不必阻塞性地等待返回值或消息，只需要向系统委托一个异步过程，那么当系统接收到返回值或消息时，系统会自动触发委托的异步过程，从而完成一个完整的流程。   
   
 5、HashMap的实现原理，HashMap是如何解决hash冲突的问题？   
 6、Redis讲一下，项目使用场景，以及对应的算法？   
 7、分布式系统的全局id如何实现？用zookeeper如何实现的呢，机器号+时间戳即可？   
 8、分布式锁的方案，redis和zookeeper那个好，如果是集群部署，高并发情况下哪个性能更好？   
 9、kafka了解么，了解哪些消息队列？   
 10、乐观锁和悲观锁   
 11、IO和NIO的却别，以及NIO的原理，有了解过mina？   
 12、JVM内存模型，JVM加载原理，回收算法了解？   
   
 二面   
   
 1、介绍你所做的项目和其中的难点？   
 2、反射的作用是什么？   
 （1）反射的主要作用是用来扩展系统和动态调用程序集。   
 （2）所谓扩展系统就是先把系统写好，系统里面定义接口，后面开发的人去写接口的代码。   
 （3）动态调用程序集就是利用反射去调用编译好的dll，当然此时的dll没有被引用到你所建的工程里面。   
 3、数据仓库，多线程和并发工具等？   
 4、私有云，docker和k8s等？   
 5、了解哪些中间件，dubbo，rocketmq，mycat等？   
 6、dubbo中的rpc如何实现？   
 7、自己实现rpc应该怎么做？   
 9、dubbo的服务注册与发现？   
 10、最后问了些排序算法   
   
 HR面   
   
 聊人生谈理想，离职原因、职业规划和期望的薪资。主要还是考察你对工作是否持积极的态度，以及你是否稳定。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第5篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴Java社招面试分享  
  
编辑于 2020-08-12 09:45:52  
  
 1、java Object有哪些方法。   
 2、然后接着问hashCode相同，equals相同？   
 答：不同，一般来说这样约定equals相等，那hashCode相等，hashCode只是对象的一个映射而已。 主要在集合中来区分不同的对象。   
   
 3、然后他接着问，那相同hashCode相等的对象在hashmap中是什么区分的？   
 答：hashmap 是由数组和链表组成的，关于hashmap的查找是根据将key做hash然后得到链表，对于hash相等的对象是通过链表保存的。查找的时候就需要迭代了。   
   
 4、如果相等hash对象太多，那么怎么解决迭代的影响？   
 5、还问了些数组与列表的区别，多线程相关的知识。   
 6、说说你对spring IOC的理解。   
 7、SpringBeanFactory 是怎么实现的？   
 答：spring通过dispathServlet或是filter加载bean，一种是基本注解的加载 一种是基本beanxml的加载。通过map来保存实例化和管理生命周期，ioc来说的话，map就相当于窗口的概念，spring 为什么能做代理，拦截器的原因。   
 可以理解为一种注册机制。这种思想还是比较常见的。   
   
 8、你对dubbo的理解，自带有哪些序列化？默认的是哪个？   
 答：dubbo也就是rmi的框架，实现方式也是将zookeeper +netty+hessian+http+webservice等东西集成在一起。 rpc的原理也就是动态代理的一个过程，那dubbo也就是将序列化和付输方式的封装然后加再加上zookeeper的协调功能。常见的序列化有xml,json,二进制。   
   
 9、zookeeper能够发生服务是否宕机或是出问题了？   
 答：zookeeper是在服务器注册时发现的，至于为什么能知道哪台服务器宕机，那肯定是做监听通过心跳这种机制了来发现服务了。但其它我觉得在zookeeper中是应该要知晓哪台服务器当前性能比较好,cpu比较低，内存使用率不高。其实我在看zookeeper的时候想过个问题。对于每一个服务来说，应当对外提供监控的功能API，就想springBoot一样有专门的模块暴露服务器的资源使用情况，zookeeper是否能监听这些数据，而不是根据随机或其它方式。   
   
 10、dubbo怎么知道服务器状态有什么内部机制？   
 11、说一下https的流程。   
 答：Https只是一种协议，将服务器客户端数据传输通过加密签名方式来传递。双方都需要安装 证书，客户端提交的数据通过加密后到服务器，服务器通过证书私钥解密，并验签，然后保证数据安全。   
   
 12、你知道https最大的重点在哪里吗？   
 答：防劫持。   
  
  
  
  
13、然后说了下  
jvm  
优化，一般你们会调整哪几个参数？一般你们设置成多少？项目中是怎么配置的？  
  
   
 答：一般会设置xmx 与xx:persize 两个参数。设置成4G xms 1G persize 。看项目情况，如果项目文件比较多，查询数据量比较大，功能复杂点的，会调高一点。 具体的值也是看情况，一般的可以看在tomcat下运行时的占用内存来看。   
   
 14、32bit 与64bit 内存设置有什么区别？   
 答：32bit 只能设置到1g多，64应该说是无限吧。   
   
 15、有一台64G的机器，设置成32的G的堆内存行不行？   
 答：不考虑业务发展的话，设这么多应该不合适吧。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第6篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴社招Java面经分享  
  
发布于 2020-08-10 08:49:10  
  
 1、jvm中一次完整的GC流程（从ygc到fgc）是怎样的，重点讲讲对象如何晋升到老年代，几种主要的jvm参数等   
 2、你知道哪几种垃圾收集器，各自的优缺点，重点讲下CMS   
 （1）Serial收集器、ParNew收集器、Parallel Scavenge收集器、Serial Old收集器、Parallel Old收集器、CMS(Concurrent Mark Sweep) 收集器、G1收集器   
   
 （2）CMS收集器是一种以获得最短回收停顿时间为目标的收集器，标记清除算法，运作过程：初始标记，并发标记，重新标记，并发清除，收集结束会产生大量空间碎片。   
   ①CMS收集器是老年代的收集器，可以配合新生代的Serial和ParNew收集器一起使用；   
   ②CMS收集器以最小的停顿时间为目标的收集器；   
   ③CMS收集器是使用“标记-清除”算法进行的垃圾回收，容易产生内存碎片   
   
 3、当出现了内存溢出，你怎么排错   
 4、JVM内存模型的相关知识了解多少   
 JVM运行时的内存结构如下：   
 （1）程序计数器：当前线程所执行的字节码的行号指示器，用于记录正在执行的虚拟机字节指令地址，线程私有。   
 （2）Java虚拟栈：存放基本数据类型、对象的引用、方法出口等，线程私有。   
 （3）Native方法栈：和虚拟栈相似，只不过它服务于Native方法，线程私有。   
 （4）Java堆：java内存最大的一块，所有对象实例、数组都存放在java堆，GC回收的地方，线程共享。   
 （5）方法区：存放已被加载的类信息、常量、静态变量、即时编译器编译后的代码数据等。（即永久带），回收目标主要是常量池的回收和类型的卸载，各线程共享   
   
 5、简单说说你了解的类加载器   
 6、JAVA的反射机制   
 7、tomcat如何调优，各种参数的意义   
 8、常见的缓存策略有哪些，你们项目中用到了什么缓存系统，如何设计的，Redis的使用要注意什么，持久化方式，内存设置，集群，淘汰策略等   
 9、如何防止缓存雪崩   
 10、分布式集群下如何做到唯一序列号   
 11、设计一个秒杀系统，30分钟没付款就自动关闭交易   
 12、如何做一个分布式锁   
 13、用过哪些MQ，怎么用的，和其他mq比较有什么优缺点，MQ的连接是线程安全的吗   
 14、MQ系统的数据如何保证不丢失   
 15、分布式事务的原理，如何使用分布式事务   
 16、什么是一致性hash   
 17、说说你知道的几种HASH算法，简单的也可以   
 18、什么是paxos算法   
 19、redis和memcached 的内存管理的区别等等

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第7篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

社招阿里Java岗面试经历  
  
发布于 2020-08-09 17:20:13  
  
 一面   
   
 1.自我介绍，包括做过项目。   
 2.有看过哪些JDK源码，了解哪些常用库。   
 3.集合框架HashMap的扩容机制，ConcurrnetHashMap的原理   
 4.jvm内存模型与gc内存回收机制   
 5.classloader结构，是否可以自己定义一个java.lang.String类，为什么？ 双亲代理机制。   
 6.了解哪些设计模式，6个设计原则分别是什么？每种设计原则体现的设计模式是哪个？   
 7.关于设计模式看了哪些书？书名是什么？   
 8.uml模型图画过哪些？ 类图中类之间的关系有哪些，区别分别是什么？   
 9.画uml中类图时候用过一种虚线么？做什么用的？   
 10.做过应用相关性能测试的，举个例子，实际项目中怎么使用的。   
 用过并发框架相关的哪些内容   
 11.了解哪些osgi的框架？   
 12有没有做过jvm内存调优，如何做的，举例子，用过哪些工具？   
     
 二面   
   
 1.自我介绍   
 2.看过哪些源码   
 3.java的io库的类结构图所用到的设计模式如何体现   
 4.画出自己设计过的设计模式如何体现，画出结构图，并进行讲解。   
 5.画出自己做的架构的项目架构图 如何扩展等   
 6.数据库设计中主键id设计的原则   
 7.jvm内存调优用过哪些工具，jstate做什么用的？如何dump出当前线程状态？   
 8.并发框架是否有了解   
 9.classloader的双亲代理机制   
 10.应用服务器的jvm调优实际经验，如何做的，在哪里用到的   
 11.在哪里获取最新资讯，逛什么论坛。最新的Swift语言有什么看法   
 12.设计原则与设计模式对应   
 13.servlet/filter作用原理配置   
 14.ibatis in操作 以及一个属性的作用   
 15.spring aop 用了什么设计原则，自动注入配置是做什么用的   
 16.jboss的类加载器   
 17.session共享机制   
 18.做过最成功的一件事情是什么？   
 19.最大的争执是什么？   
 20.为什么想要离职去阿里   
     
 三面   
   
 1.现在公司负责什么？   
 2.项目主要目的是做什么的？   
 3.公司管理方式、项目问题反馈机制是什么？   
 4.Java 的序列化做什么用的 序列化id会出现哪些问题？   
 5.OSGi用过哪些？类加载器结构如何，如何在一个bundle中加载另外一个bundle中的一个类？   
 6.nio是否了解 阻塞之后通知机制是怎样的？   
 7.uml设计类图如何画，类之间关系以及区别   
 8.spring如何不许要配置文件加载bean定义，可能是问自动注解或者是properties文件定义bean   
 9.ibatis等框架是不是都是实际在使用的,技术细节   
 10.为什么想离职？   
     
 四面   
   
 1.自我介绍   
 2.公司做什么，业务， 负责内容，汇报机制   
 3.企业级应用安全相关   
 4.http协议，返回码，301与302区别   
 5.多线程并发用过哪些   
 6.应用服务器相关，谈最熟悉的   
 7.为什么离职

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第8篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里社招Java面试经历  
  
发布于 2020-08-08 11:15:16  
  
 一面   
   
 1、自我介绍，讲讲求职意向和擅长的技能。   
 2、Tomcat的基本架构是什么？   
 Server、Connection和Container（Engine）、Service、Context、Host、Wrapper、Container、LifeCycle、Executor、Bootstrap和Catalina   
     
 3、JVM调优，工作中怎么用，GC算法和回收策略有哪些？   
 4、Java中线程池是如何实现的？   
 5、创建线程池的几个核心构造参数是什么？   
 corePoolSize（核心线程数）、queueCapacity（任务队列容量）、maxPoolSize（最大线程数、keepAliveTime（线程空闲时间）、allowCoreThreadTimeout（允许核心线程超时）、rejectedExecutionHandler（任务拒绝处理器）   
     
 6、Volatile和Synchronize的区别？   
 （1）Volatile轻量级的，只能修饰变量。synchronize重量级的，还可以修饰方法   
 （2）Volatile只保证数据的可见性，不能用来同步，因为多线程访问Volatile变量不会阻塞   
 （3）synchronize不仅保证可见性，而且保证原子性，因为自由获得了锁的线程才能到达临界区，从而保证了临界区中的所有语句被执行，多个线程抢夺synchronize锁的时候，会出现阻塞。   
     
 7、GC的机制是什么？垃圾回收器的基本原理是还说呢么？是否可以立即回收内存？怎么样主动的通知JVM进行垃圾回收？   
 8、类加载器双亲委派模型机制？   
 9、集合的种类和区别，HashMap底层如何实现(JDK1.8有所改动)，HashMap和ConcurrentHashMap的区别？   
 10、Spring IOC如何管理Bean之间的依赖关系，怎么样避免循环依赖？   
 11、SpringBean创建过程中的设计模式？   
     
 二面   
   
 1、自我介绍。   
 2、InnoDB支持的四种事务隔离级别名称是什么？ 之间的区别是什么？   
 3、聊一聊事务的特性   
 4、什么是MySQL隔离级别？   
 5、如何理解BTree机制？   
 6、谈一谈对慢查询的分析？MySQL常用的优化方法有哪些？   
 7、谈一谈悲观锁和乐观锁以及SQL的实现。   
 8、简述三次握手和四次握手的过程？   
 9、B+树索引和Hash索引的区别？   
  
  
  
 三面   
   
 1、自我介绍。   
 2、MySQL的锁并发？   
 3、高并发场景下如何防止死锁，保证数据的一致性？   
 4、集群和负载均衡的算法与实现？   
 5、简述分库与分表设计？   
 6、分库分表带来的分布式困境与对应之策有哪些？   
 7、加锁的机制是什么？   
 8、Redis和Setnx命令使如何实现分布式锁的？使用Redis怎么进行异步队列？会有什么缺点？

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第9篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里Java社招面试经历分享  
  
编辑于 2020-08-07 10:24:07  
  
 1、java并发基础：volitate关键字原理，AQS原理，线程池关键参数&原理，concurrenthashmap原理；   
 2、java虚拟机：java栈和堆内存分布，垃圾回收原理，常见的垃圾回收算法，常见的stackoverflowexception,outofmemoryexception是怎么回事；   
 3、消息中间件：消息中间件解耦&消峰特性的了解，消息中间件的原理，事务消息二阶段提交的原理；   
 4、远程服务框架:dubbo的通信过程，服务发现过程，序列化反序列过程原理理解；   
 5、分布式协调算法：zk的paxos协议原理,zab原理，metaq的raft协议原理；   
 6、数据库mysql:innodb的索引原理、事务原理、锁的原理和实现、事务隔离级别mvcc原理，innodb和myisam的区别；   
 7、nosql系列：redis的核心api需要熟悉，大概的原理，如何解决缓存被击穿的问题，如何解决缓存热key？   
 8、分布式锁的实现原理和有多少种实现方式？   
 目前主流的分布式锁的实现方式有三种：   
 ①借助数据库来实现，新建一张锁表；   
 操作前向表添加一条锁记录（锁id建立唯一索引），成功添加者获得锁权限，处理完后删除锁记录来释放锁。   
   
 ②基于缓存实现，如memcache 和 redis；   
 memcache的add操作具有原子性，可以保证同一个key add操作只有一个成功，来获取锁权限，利用缓存的失效时间来解决死锁问题。相对于第一种方案，这种方案性能更好，而且操作更方便。   
   
 ③通过zookeeper实现；   
 客户端会在zookeeper生成一个临时的目录节点，存储在一个序列中，每次节点序号最小的节点对应的客户端获得锁，处理完成后删除最小节点，而且可重复获取锁（通过判断序号是否和最小的节点相同）。这种方式可以实现阻塞分布式锁，和锁的重复获取问题。   
   
 9、分布式限流如何实现，原理是什么？   
 10、运维命令：如何排查cpu100%，如果机器内存出了问题，应该要怎么排查？   
 11、分库分表的实现原理是什么，你所在业务一般是怎么分库分表的？对应逻辑是什么？   
 12、核心一些中间件的高可用是如何保证的？体系化看下；

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第10篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴Java岗社招面试经历  
  
编辑于 2020-08-07 08:51:08  
  
 1、列举几个Java Collection类库中的常用类 ；   
   
 2、List、Set、Map是否都继承自Collection接口？   
 3、HashMap和Hashtable的区别；   
 主要的区别有：线程安全性，同步，以及速度。   
 （1）HashMap几乎可以等价于Hashtable，除了HashMap是非synchronized的，并可以接受null(HashMap可以接受为null的键值(key)和值(value)，而Hashtable则不行)。 HashMap是非synchronized，而Hashtable是synchronized，这意味着Hashtable是线程安全的，多个线程可以共享一个Hashtable；而如果没有正确的同步的话，多个线程是不能共享HashMap的。Java 5提供了ConcurrentHashMap，它是HashTable的替代，比HashTable的扩展性更好。   
   
 （2）另一个区别是HashMap的迭代器(Iterator)是fail-fast迭代器，而Hashtable的enumerator迭代器不是fail-fast的。所以当有其它线程改变了HashMap的结构（增加或者移除元素），将会抛出ConcurrentModificationException，但迭代器本身的remove()方法移除元素则不会抛出ConcurrentModificationException异常。但这并不是一个一定发生的行为，要看JVM。这条同样也是Enumeration和Iterator的区别。 由于Hashtable是线程安全的也是synchronized，所以在单线程环境下它比HashMap要慢。如果你不需要同步，只需要单一线程，那么使用HashMap性能要好过Hashtable。 HashMap不能保证随着时间的推移Map中的元素次序是不变的。   
   
 4、HashMap中是否任何对象都可以做为key,用户自定义对象做为key有没有什么要求？   
 用自定义类作为key，必须重写equals()和hashCode()方法。   
 自定义类中的equals() 和 hashCode()都继承自Object类。   
 Object类的hashCode()方法返回这个对象存储的内存地址的编号。   
 而equals()比较的是内存地址是否相等。   
   
 5、interface 和 abstrat class的区别 是否可以继承多个接口，是否可以继承多个抽象类   
 6、 启动一个线程是用run()还是start()?？   
 7、多线程有几种实现？同步和并发是如何解决的？   
 8、什么叫守护线程，用什么方法实现守护线程？如何停止一个线程？   
 9、解释是一下什么是线程安全？举例说明一个线程不安全的例子。   
 10、解释Synchronized关键字的作用。   
 11、当一个线程进入一个对象的一个synchronized方法后，其它线程是否可进入此对象的其它方法？   
 12、了解过哪些JDK8的新特性，举例描述下相应的特性？   
 13、对sql进行优化的原则有哪些？   
 （1）减少返回不必要的数据   
 （2）减少物理和逻辑读次数   
 （3）减少计算次数   
   
 14、servlet生命周期是生命与cgi的区别？   
 （1）CGI（Common Gateway Interface通用网关接口）程序来实现数据在Web上的传输，使用的是如Perl这样的语言编写的，它对于客户端作出的每个请求，必须创建CGI程序的一个新实例，这样占用大量的内存资源。由此才引入了Servlet技术。   
 （2）Servlet提供了Java应用程序的所有优势——可移植、稳健、易开发。使用Servlet Tag技术，Servlet能够生成嵌于静态HTML页面中的动态内容。   
   
 15、StringBuffer有什么优势？为什么快？

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第11篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴后端Java社招面试分享  
  
编辑于 2020-08-06 14:45:59  
  
 1、String和StringBuffer的区别   
 （1）运行速度：StringBuilder >StringBuffer >String   
 String是字符串常量，不可变，每次改变只是创建一个新的对象，然后GC回收掉老的那个，所以执行速度最慢，另外两个是字符串对象，可变。   
 （2）线程安全： StringBuilder是线程不安全的，StringBuffer是线程安全的，看是否带synchronized关键字。多线程则采用StringBuffer，单线程则要建议用速度较快的StringBuilder。   
 （3）String：适用于少量的字符串操作的情况，String是final类，无法被继承。StringBuilder：适用于单线程下在字符缓冲区进行大量操作的情况。 StringBuffer：适用多线程下在字符缓冲区进行大量操作的情况。   
   
 2、gc的概念，如果A和B对象循环引用，是否可以被GC？   
 答：这个循环引用是否被回收，就看这个循环引用是否挂在根上，A引用B，B引用A，A和Ｂ并没有挂在某个内存元和根上，当他们的生命周期结束的时候。这两个对象都有可能被回收。   
   
 4、String s = “123”;这个语句有几个对象产生？   
 5、Error、Exception和RuntimeException的区别，作用又是什么？   
 Error是Throwable 的子类，用于指示合理的应用程序不应该试图捕获的严重问题。大多数这样的错误都是异常条件。虽然 ThreadDeath 错误是一个“正规”的条件，但它也是 Error 的子类，因为大多数应用程序都不应该试图捕获它。在执行该方法期间，无需在其 throws 子句中声明可能抛出但是未能捕获的 Error的任何子类，因为这些错误可能是再也不会发生的异常条件。   
 Exception类及其子类是 Throwable 的一种形式，它指出了合理的应用程序想要捕获的条件。   
 RuntimeException是那些可能在 Java 虚拟机正常运行期间抛出的异常的超类。可能在执行方法期间抛出但未被捕获的RuntimeException 的任何子类都无需在 throws 子句中进行声明。它是Exception的子类。   
   
 7、reader和inputstream区别；   
 （1）InputStream是表示字节输入流的所有类的超类；Reader是用于读取字符流的抽象类   
 （2）InputStream提供的是字节流的读取，而非文本读取，这是和Reader类的根本区别。 即用Reader读取出来的是char数组或者String ，使用InputStream读取出来的是byte数组。   
   
 8、hashCode的作用；   
 hashCode方法的主要作用是为了配合基于散列的集合一起正常运行，这样的散列集合包括HashSet、HashMap以及HashTable。   
 Java中的hashCode方法就是根据一定的规则将与对象相关的信息（比如对象的存储地址，对象的字段等）映射成一个数值，这个数值称作为散列值。   
   
 9、Object类中有哪些方法，列举3个以上；   
 10、char型变量中能不能存贮一个中文汉字？为什么？   
 char型变量是用来存储Unicode编码的字符的，unicode编码字符集中包含了汉字， 所以，char型变量中当然可以存储汉字。不过，如果某个特殊的汉字没有被包含在 unicode编码字符集中，那么，这个char型变量中就不能存储这个特殊汉字。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第12篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴社招Java面试经历  
  
编辑于 2020-08-06 13:19:20  
  
 1、谈谈你对HashMap的理解，底层原理的基本实现，HashMap怎么解决碰撞问题的？   
 2、开发中用了哪些数据库？回答mysql，储存引擎有哪些？   
 3、然后问悲观锁和乐观锁问题使用场景、分布式集群实现的原理。   
 4、springmvc和mybatis的工作原理，有没有看过底层源码？   
 5、熟悉IO么？与NIO的区别，阻塞与非阻塞的区别   
 6、微信红包怎么实现？   
 7、海量数据分析。   
 8、线程安全和非线程安全。   
 9、HTTP2.0、thrift。   
 10、java反射应用   
 11、分布式事务一致性。   
 12、nio的底层实现。   
 13、jvm基础是必问的，jvm GC原理，JVM怎么回收内存。   
 14、API接口与SDI接口的区别？   
 15、dubbo如何一条链接并发多个调用。Dubbo的原理，序列化相关问题。   
 16、用过哪些中间件？   
 17、做过工作流引擎没有。   
 18、以前的工作经历，自己觉得出彩的地方 。   
 19、线程池的一些原理，锁的机制升降级 。   
 20、从系统层面考虑，分布式从哪些纬度考虑 ？   
 21、Hadoop底层怎么实现 ？   
 22、threadLocal，线程池，hashMap/hashTable/coccurentHashMap等 。   
 23、秒杀系统的设计 。   
 24、虚拟机，IO相关知识点 。   
 25、Linux的命令 。   
 26、一个整形数组，给定一个数，在数组中找出两个数的和等于这个数，并打印出来，还要求O(n)。   
 27、n个整数，找出连续的m个数加和是最大。   
 28、数据库锁隐原理 。   
 29、1000个线程同时运行，怎么防止不卡？   
 30、并列的并发消费问题。   
 31、高并发量大的话怎么处理热点，数据等？   
 32、如何获取一个本地服务器上可用的端口？   
 33、流量控制相关问题。   
 34、数据库TPS是多少，是否进行测试过？   
 35、缓存击穿有哪些方案解决？   
 36、Java怎么挖取回收器相关原理？   
 37、Java的集合都有哪些，都有什么特点？   
 38、分布式锁，redis缓存，spring aop，系统架构图，MySQL的特性；   
 39、场景：同时给10万个人发工资，怎么样设计并发方案，能确保在1分钟内全部发完？   
 40、Java中的内存溢出是如何造成的？   
 （1）内存中加载的数据量过于庞大，如一次从数据库取出过多数据；   
 （2）集合类中有对对象的引用，使用完后未清空，使得JVM不能回收；   
 （3）代码中存在死循环或循环产生过多重复的对象实体；   
 （4）使用的第三方软件中的BUG；   
 （5）启动参数内存值设定的过小；

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第13篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里蘑菇街和其他两个小公司社招面经分享（Java）  
  
发布于 2020-08-06 10:42:43  
  
 阿里巴巴   
   
 1、首先让我做了下自我介绍，然后问了我几个基本的问题   
 2、hashMap和concurrentHashMap的区别，内部实现原理；   
 3、stringBuild和stringBuffer的区别等；   
 4、然后让写一个单例模式的实现   
 5、然后基本上看着简历问了，做过项目吗？在项目中解决了哪些问题？看过开源代码吗？   
 6、struct2的工作原理是什么？   
 7、然后问jvm的内存结构有哪些？   
 8、Linux写过shell吗？能直接写一下吗？   
 9、你理解的高可用是什么，能实现秒杀吗？   
     
 蘑菇街   
   
 1、首先让自我介绍，然后主要看着简历中写到的东西提问。   
 2、负载均衡的处理有哪些方式？   
 3、lvs了解吗？你们为什么从iphash转到了使用最小链接的负载算法？平台的用户总量。   
 4、mq的p2p模式，阿里的中间件技术了解吗？   
 5、然后问为什么从memcach迁到了redis，使用redis中遇到了哪些问题？   
     
 其他两家小公司   
   
 1、concurrentHashMap内部结构   
 2、ajax请求的封装对象、负载均衡的几种策略；   
 3、redis设置过期时间的实现原理，redis的并发操作数据怎么保证安全性？   
 4、Nginx轮询的实现原理；pring的事物及实现模式；   
 5、数据库优化的策略，索引的数据结构；   
 6、springMVC的工作原理，常用的设计模式；   
 7、ajax异步刷新实现的步骤，如何判断是Ajax请求？   
 8、servlet的生命周期，怎么判断是get请求还是post请求？   
 9、后台实现session共享的方法；过滤器和拦截器的区别。   
 10、同花顺的搜索实现方案；   
 11、Linux命令直接搜索大文档中的几行内容，按热度实现推荐算法的方案，单例模式获取到两个项实例的方案；   
 12、分库分表实践，Nginx轮询实现原理（组件的实现），序列化的原理，分布式一致性解决方案。   
 13、redis的锁机制，treeMap实现原理，一致性hash原理，同步锁；   
 14、TCP/UDP的区别，说一下TCP的三次握手（建立连接）和四次挥手（四次挥手）。   
 15、Java8的一些新特性介绍。项目中遇到的问题，印象最深的问题！   
 16、线程进程的区别，Nginx负载均衡、LVS负载均衡；   
 17、对springBoot理解，spring的增强方式；   
 18、AOP，ZK保证一致性的方法；   
 19、分布式事务的实现方式，分布式锁，分布式一致性，redis分布式锁；

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第14篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴后端Java社招面经  
  
编辑于 2020-08-04 11:12:11  
  
 一面   
   
 1、先自我介绍，也讲了一些有关日常的工作   
 2、父子线程怎么共享数据 interitableThreadLocals；   
 3、HashMap1.7、1.8区别，lock和sync区别；   
 4、AQS原理，执行过程源码，入队出队的细节，源码细节；   
 5、CountDownLatch和CyclicBarrier的区别是什么？   
 6、数据库方面，mysql索引，问各种树的特性；   
 7、举例优化sql、MVCC和事务隔离级别的关系、间隙锁、行锁；   
 8、可达性分析算法中根节点有哪些、cms和G1区别、怎样GC调优、怎样排查CPU彪高、内存彪高、逃逸分析；   
 9、redis数据结构、跳跃表、redis qps能上多少，怎么知道的？   
 10、sentinel和cluster区别和各自适用场景；   
 11、redis cluster集群同步过程；   
 12、redis单线程为什么快？   
 13、热key产生原因和后果以及怎么解决？   
 14、本地缓存需要高时效性怎么办？   
 15、spring的作用、spring循环依赖怎么解决？spring aop原理，spring bean生命周期；   
 16、dubbo服务暴露和引用过程，负载均衡策略，容错机制在哪里实现的源码   
 17、为什么换工作？   
   
 二面   
   
 1、自我介绍、项目介绍   
 2、让设计一个防止缓存穿透的解决方案，简单的就是存null值，可以结合布隆过滤器，设计分布式系统；   
 3、然后又会问到流量分发到具体过滤器服务的方式，比如一致性hash算法，怎么调用？比如dubbo直连等等   
 4、有没有做过限流？设计一个侵入性最小的限流服务。   
 5、项目中碰到的问题   
 6、为什么换工作？   
   
 三面   
   
 1、项目介绍   
 2、dubbo服务调用过程   
 3、dubbo调用端怎么在jvm中生成对应服务？dubbo服务端和调用端超时时间设置和区别、dubbo长连接。   
 4、NIO、BIO区别，NIO解决了什么问题，Netty线程模型（源码拷问）。   
 5、MQ相关（RocketMQ、kafaka）   
 6、mysql行锁最大并发数？   
 7、设计秒杀系统，我说用异步的方式，会问怎么优化？改为了同步的方式，异步和同步区别？   
 8、碰到哪些技术难点？怎么解决？有没有参考其他大厂？其他大厂方案什么样的？   
 9、刚刚的秒杀系统，会涉及到多个库表的更新，分布式事务怎么解决？异步？有没有更好的方案？同步TCC方式，TCC方式原理？

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第15篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴Java社招面经分享  
  
编辑于 2020-08-04 10:36:24  
  
 一面   
   
 1、自我介绍；   
 2、介绍自己做的项目；   
 3、说一说项目中最有难度和记忆最深刻的项目；   
 4、说说Hashmap 原理；   
 5、底层红黑树是什么？   
 6、ArrayList 与 LinkedList的区别？   
 7、ArrayList 扩容的过程是什么样子的？   
 8、常用排序算法和时间复杂度；   
 9、快排的实现原理：双指针，建议在纸上自行手写实现，这样才容易记牢；   
 10、Kafka 事务处理机制；   
 11、Kafka 消息重复消费如何处理？   
 12、Kafka 如何保证消息的可靠性？   
 13、线上FullGC 处理方法；   
 14、垃圾回收介绍一下；   
 15、多进程和多线程有什么区别？   
 16、类加载机制；   
 17、开发过程中关于规范的代码有什么好的可以分享？   
 18、Spring 解决循环依赖；   
   
 一面相对来说是比较偏技术细节的，从语言本身到 JVM，再到框架和工具都有涉及。   
   
 二面   
   
 1、介绍自己，并介绍一下自己做的项目？   
 2、自己主要负责哪块内容，其中印象最深刻的项目是什么？   
 3、项目中遇到的最难的问题是什么，怎么解决的？   
 4、项目在生产上有没有遇到过问题，是什么样子的问题以及如何解决的？   
 5、有没有因为 bug 或者其他原因导致的线上问题？   
 6、HashMap 与 ConcurrentHashMap的区别？   
 7、线程池的实现原理；   
 8、说一下 JVM 有哪些垃圾回收器？   
 9、详细介绍一下 CMS 垃圾回收器？   
 10、新生代垃圾回收器和老年代垃圾回收器都有哪些？它们有什么区别？   
 11、简述分代垃圾回收器是怎么工作的？   
 12、GC 垃圾回收器的理解；   
 13、对SQL 优化有什么建议？   
 14、Kafka 的基本组件和消息流程；   
 15、使用 Redis 如何实现一个分布式锁？   
 16、你觉得工作这么久，自己对业务有什么重大的贡献？   
 17、项目小组有多少人，个人觉得自己在小组中是什么样的位置？   
 18、工作这么久觉得业务上有什么缺陷，怎么优化？   
 19、是否有参与项目架构的讨论和设计？   
 20、业余时间一般做些什么？   
 21、既然你有 Github 账号，那么你GitHub 上印象最深刻的项目是什么？   
   
 二面考察的是各方面的能力，包括项目经历、设计能力、沟通能力以及学习能力。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第16篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

风云科技/支付宝/宏晶信息科技后端Java社招面经  
  
编辑于 2020-08-04 15:28:15  
  
 风云科技   
   
 面试过程：   
 1、自我介绍之前的工作经历以及为什么辞职原因，自己为什么符合这一家公司，未来几年的计划是什么，对自己有什么要求，是否会长期呆在这个城市，对公司了解多少？   
 2、引用传递和值传递。   
 3、sso单点登陆怎么控制权限；（shiro）   
 4、webservice接口。   
 5、分布式事务；（两个本地事务外加消息中间件解决分布式事务的数据一致性）   
 6、多线程怎么通信？   
   
 支付宝   
   
 面试过程：   
 1、先自我介绍，没有直接问项目相关的，而是直接问的是简历上所写的各个技术点；   
 2、有多线程，线程池，数据库索引等相关的问题。   
 3、技术面完后，就是聊项目，会详细问项目中的各个关键点，实现方式，用到技术点的实现原理。比如我的项目中用到Netty，那么就会问到Netty的各个核心模块，NIO等询问起。   
 4、J.U.C锁和monitor之间的区别，实现，什么时候该用哪种锁，为什么？数据库各种锁什么时候用？   
 5、数据库创建索引的时候会锁表吗？为什么？   
 答：会。因为索引用的是B+ 树，要创建好索引，那么必须要建立好树结构。   
   
 宏晶信息科技   
   
 面试过程：   
 1、先让我自我介绍，稍微讲了一下项目经历和离职原因   
 2、java中的封装，动态绑定   
 3、也有一些数据库的使用，sql优化，shell命令   
 4、JVM的内存管理和长sql query的优化。   
 答：JVM就说一说以前的三层模型，以及GC的工作原理。SQL主要就是组合index的选用。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第17篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里Java社招面经  
  
发布于 2020-08-03 08:35:14  
  
 一面   
   
 1、常见集合类的区别和适用场景   
 2、并发容器了解哪些？   
 3、如何判断链表是否有环   
 4、concurrentHashMap如何实现   
 5、集群服务器 如何application 共享   
 6、JAVA网络编程中：BIO、NIO、AIO的区别和联系   
 7、jvm内存模型jmm 知道的全讲讲   
 8、JAVA的垃圾回收，标记算法和复制算法的区别，用在什么场合？   
 9、http和https的区别，http1.x和http2.0的区别，SSL和TSL之间的区别   
 10、GC、G1和ZGC的区别   
 11、B+树和B树的区别，和红黑树的区别   
 12、内存泄漏与内存溢出的区别   
 13、session的生命周期是多久   
 14、关于Mina框架了解多少？（因为我在项目里用到了Mina，所以提到了这个部分）   
     
 二面   
   
 1、java cas原理   
 2、JAVA线程池有哪些参数，如果自己设计一个线程池要考虑哪些问题？   
 3、Java的lock的底层实现？   
 4、mysql数据库默认存储引擎，有什么优点   
 5、MySQL的事务隔离级别，分别解决什么问题。   
 6、四个表，记录成绩，每个大约十万条记录，如何找到成绩最好的同学   
 7、常见的负载均衡算法有哪些   
 8、如果Redis有1亿个key，使用keys命令是否会影响线上服务   
 9、Redis的持久化方式，aod和rdb，具体怎么实现，追加日志和备份文件，底层实现原理的话知道么   
     
 三面   
   
 1、请画一个完整大型网站的分布式服务器集群部署图   
 2、多个RPC请求进来，服务器怎么处理并发呢   
 3、讲一下Redis的哨兵机制   
 4、数据库分库分表一般数据量多大才需要？   
 5、如何保证数据库与redis缓存一致的   
 6、项目中消息队列怎么用的？使用哪些具体业务场景？   
 7、JVM相关的分析工具有使用过哪些？具体的性能调优步骤吗？   
 8、MySQL的慢sql优化一般如何来做？除此外还有什么方法优化？   
 9、线上的服务器监控指标，你认为哪些指标是最需要关注的？为什么？   
 10、如何做压测，抗压手段   
 11、秒杀模块怎么设计的   
     
 HR面   
   
 1、自我介绍   
 2、你怎么评价你之前的3轮面试？   
 3、你怎么看待你自己，你最大的核心竞争力是什么？   
 4、未来自己的职业规划；   
 5、对阿里技术氛围有什么样的理解，用过哪些阿里的开源库   
 6、期望的薪资是多少？   
 7、最后，你有什么想了解的？

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第18篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴社招Java岗面筋  
  
发布于 2020-07-29 10:42:17  
  
 一面   
   
 1、先是自我介绍，介绍自己的项目经验和个人的擅长点，面试官主要考察你的表达能力和语言精简能力。   
 2、项目中做了什么，难点在哪里？主要是想了解参与过技术难度最大的项目难点，技术难点在哪里，你是怎么来解决的，考察项目经验(技术难度)。   
 3、Java的线程池说一下，各个参数的作用，如何进行的？   
 4、同步与异步区别？   
 5、HashMap的实现原理，HashMap是如何解决hash冲突的问题？   
 6、Redis讲一下，项目使用场景，以及对应的算法？   
 7、分布式系统的全局id如何实现？用zookeeper如何实现的呢，机器号+时间戳即可？   
 8、分布式锁的方案，redis和zookeeper那个好，如果是集群部署，高并发情况下哪个性能更好？   
 9、kafka了解么，了解哪些消息队列？   
 10、乐观锁，悲观锁；   
 11、IO和NIO的区别，以及NIO的原理，有了解过mina？   
 12、JVM内存模型，JVM加载原理，回收算法了解？   
 13、你是想做业务还是研究？   
   
 二面   
   
 1、介绍一下你做的项目和其中的难点；   
 2、反射的作用是什么？   
 3、数据仓库，多线程和并发工具等？   
 4、私有云，docker和k8s等？   
 5、了解哪些中间件，dubbo，rocketmq，mycat等？   
 6、dubbo中的rpc如何实现？   
 7、自己实现rpc应该怎么做？   
 8、dubbo的服务注册与发现？   
 9、最后问了些排序算法；   
   
 三面   
   
 三面不是面试，而是笔试，耗时三个小时，考的是Java核心的基础。   
 1、Java并发的知识点   
 2、集合类、线程池   
 3、多线程之间的通信   
   
 HR面   
   
 1、聊人生谈理想，主要还是考察你对工作是否持积极的态度，以及你是否稳定。   
 2、询问一下从上家公司离职，选择阿里的原因是？   
 3、对于未来自己有着怎样的规划？   
 4、对薪资的期望是什么？   
   
 总结   
 阿里的面试特别喜欢面试技术原理，特别是：   
 1、多线程、NIO、异步消息框架分布式相关的缓存算法等；   
 2、JVM的加载过程和原理、回收算法、以及具体使用过的框架，会问部分参数检验你是否熟用；

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第19篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

思谷软创、阿里巴巴Java工程师社招面试经历  
  
编辑于 2020-07-28 19:49:15  
  
 思谷软创   
   
 面试过程：   
 1、介绍下你以前做过的项目,用到了哪些技术，项目的业务背景和优势是什么？   
 2、SSH，MVC的概念   
 3、jdk底层的算法和网络相关   
 4、JVM的相关问题，并发的相关类的实现机制   
 5、object有哪几个方法，分别是干什么用的？   
 答： wait，notify，equls，tostring……一共九个   
   
 6、哈希map的实现，如何避免哈希冲突？   
 7、一个程序运行的全过程、线程池，classLoader   
 8、spring、hibernate、数据库   
 9、分布式部署的系统中当第1次访问时，在A服务器所部署的系统中的session中存个东西，当第二次访问时，跑到B服务器所部署的系统中去取这个东西时取不到了怎么办？   
 答：用缓存技术代替session。   
   
 10、离职的原因和薪资期望；   
   
 阿里巴巴   
   
 一面：   
 开发基础：数据结构、事务隔离级别、技术平台、开发框架、项目细节、分布式事务、分布式锁、Java并发模型、JVM、缓存，问的广，偏理论。   
   
 二面：   
 1、项目经验和对中间件实现的理解；   
 2、dubbo的优化；   
 3、线程问题排查；   
 4、jvm优化；   
 5、redis过期键处理；   
 6、分布式事务实现；   
 7、hashMap的底层实现   
   
 三面：   
 1、业务理解和架构，技术平台的选择、架构设计的过程和考量；   
 2、个人发展定位、对技术点的预判；   
 问题开放，主要看你个人的视野、架构能力、定位、大局观；   
   
 四面：（HR面）   
 问职业经历、工作组成、工作评价和个人预期；

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第20篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

支付宝、帜讯信息后端Java社招面经  
  
编辑于 2020-07-28 17:23:16  
  
 支付宝   
   
 面试过程：   
 1、工作中做的最有成就的事情，遇到的最困难和解决方案。问的很深，比如我说序列化速度慢，使用fastjson转换就快了，他就会问为什么以前慢，fastjson快在哪里。   
 2、hsmap的实现原理。（只知道遍历用iterator，还有红黑树那个结构）   
 3、多线程的安全性。   
 4、大数据量并发；   
 5、调用远程服务。（我只用dubbo实现过一个demo）   
 6、dfs。（我的理解就是存储和获取二进制文件的服务器；）   
 7、spring源码， aop和ioc，beanfactory的源码以及从webcontex到bean的继承关系。   
 8、数据库连接；   
 9、spring mvc的原理，说说一个请求开始到结束共有几个步骤。   
 10、TreadLocal是做什么用的？   
   
 帜讯信息   
   
 笔试：   
 servlet 相关的内容，生命周期、过滤器等；   
 并发编程相关   
 CSS 盒子模式，页面加载   
 javascript 的方式，javascript 中 this 的含义，常见的 HTTP 请求头等等；   
 数据库相关：事务的级别、传播方式，like 模糊查询的优化。   
   
 面试：   
 1、自我介绍，工作经历和项目的介绍，在工作的期间学习了哪些知识，工作中用到了哪些测试工具？   
 2、JavaScript的基本数据类型？   
 答:由于Javascript中的变量是松散类型的，所以它提供了一种检测当前变量的数据类型的方法，也就是typeof关键字，通过typeof关键字，对这5种数据类型会返回下面的值：   
 以字符串形式显示 “undefined” ---------- 如果值未定义 Undefined “boo lean” ---------- 如果这个值是布尔值 Boolean “string” ---------- 如果这个值是字符串 String “number” ---------- 如果这个值是数值类型 Number “object” ---------- 如果这个值是对象或null Object “function” ---------- 如果这个值是函数 Function；

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第21篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# 【Fake面经】三年网易、滴滴、点我Java岗面试经验汇总

精  
  
  
编辑于 2020-07-12 20:32:28  
  
  
 女朋友约会催的紧，先发一部分，周日晚补齐   
  
  
 对回答有疑惑或者有想交流的同学直接留言或者私戳我都可以   
  
  
  
  
  
原始面经链接：  
https://www.nowcoder.com/discuss/438830?type=2&channel=2000&source\_id=discuss\_center\_discuss\_jinghua  
  
 以下是Mock回答   
 回答由三部分组成   
  
 直接回答：会用简要的语言叙述这个问题的答案   
 引申点（选择性给出）：该点是预测面试官会感兴趣or会问的其他要点，可以当成进阶知识掌握   
 业务相关（选择性给出）：该部分是在真实业务中遇到过的关于这个问题的处理和思考   
  
 核心思路是在基础扎实的回答之上尽可能的扩散出自己深度和广度来，从业务和类似技术来举一反三抢占话语权   
 面经   
 Java基础   
 0.HashMap的源码，实现原理，JDK8中对HashMap做了怎样的优化。   
 拉链结构，数组+链表，原理是hash找数组，冲突后拉链表，1.8优化为会进化成红黑树提高效率，并且使用2^n来做容量值   
  
 引申点：   
  
 equal & hashcode   
 其他地方的hash处理，如redis的hash、集群slot等   
 对hash算法类型的了解（安全哈希和非安全哈希如mermerhash）   
 对hashMap实现的了解：取hashcode，高位运算，低位取模   
 一致性hash（处理了什么问题，在什么场景用到）   
 红黑树简单描述   
  
  
 1.HaspMap扩容是怎样扩容的，为什么都是2的N次幂的大小。   
 在容量到达抵达负载因子\*最大容量的时候进行扩容，负载因子的默认值为0.75   
 2N的原因：   
  
 hash的计算是通过hashcode高低位混合然后和容量的length进行与运算   
 在length=2n的时候，与运算相当于是一个取模操作   
 那么在每次rehash完毕之后mod2N的意义在于要么该元素是在原位置，要么是在最高位偏移多一位的位置，提高效率   
  
  
 引申点：   
  
 ConcurrentHashMap的扩容：1.7分段扩容以及1.8transfer并发协同的扩容   
 redis渐进式hash扩容处理   
  
   
  
  
   
  
 3.HashMap，HashTable，ConcurrentHashMap的区别。   
 Map线程不安全（没有用任何同步相关的原语），Table安全（直接加syn），Concurrent提供更高并发度的安全（分段锁思想orSyn+Cas）   
  
 引申点：   
  
 对线程安全的定义：如hashmap在1.7前会头插死循环，但是在1.8改善后还是不能叫线程安全，因为没有可见性   
 对锁粒度的思考：在介于map和table之间存在tradeoff之后的均衡解   
 Syn和ReentranceLock的区别   
 锁升级   
  
  
 4.极高并发下HashTable和ConcurrentHashMap哪个性能更好，为什么，如何实现的。   
 分两种情况讨论：   
  
 极高并发读：并发读的情况下，Table也为读加了锁，没有并发可言，ConcurrentMap读锁并没有加并发，直接可读，若读resize的某个tab为空则转到新tab去读，Node的元素val和指针next都是volatile修饰的，可以保证可见性，所以concurrentMap获胜   
 极高并发写：在并发写的情况下，table也是直接加了Syn做锁，强制串行，并且resize也只能单线程扩容，ConcurrentMap首先对于每个数组都有并发度，其次在resize的时候支持多线程协同，所以concurrentMap获胜   
  
 所以整体而言concurrentMap优势在于：   
  
 读操作基于volatile可见性所以无锁   
 写操作优势在于一是粗粒度的数组锁，二是协同resize   
  
 这个问题的思路是先分类讨论然后描述细节最后在下结论   
  
 引申点：   
  
 volatile的实现：保证内存可见、禁止指令重排序但无法保证原子性   
 java内存模型   
 JVM做的并行优化、先行发生原则与指令重排序   
 底层细节的熟悉   
  
  
 5.HashMap在高并发下如果没有处理线程安全会有怎样的安全隐患，具体表现是什么。   
 1.7前死锁，1.7后线程会获取脏值导致逻辑不可靠   
 6.java中四种修饰符的限制范围。   
 public：公用，谁来了都给你用   
 protected：包内使用，子类也可使用   
 default：包内使用，子类不可使用   
 private：自己用   
 7.Object类中的方法。   
 wait\hashcode\equal\wait\notify\getclass\tostring\nofityall\finalize   
  
 引申点：   
  
 wait和sleep区别   
 hashcode存在哪儿（对象头里）   
 finalize作用：GC前执行，但是不一定能把这个函数跑完   
 getClass后能获取什么信息：引申到反射   
  
  
 8.接口和抽象类的区别，注意JDK8的接口可以有实现。   
 接口：可以imp多个接口，1.7之前不允许实现，1.8后可以实现方法   
 抽象类：只能继承一个类，抽象类中可以存在默认实现方法   
 接口的语义是继承该接口的类有该类接口的行为   
 抽象类的语义是继承该抽象类的类本身就是该抽象类   
 9.动态代理的两种方式，以及区别。   
  
 CGLIB：其本质是在内存中继承了一个子类，可以代理希望代理的那个类的所有方法   
 JDK动态代理：实现InvocationHandler，通过生成一个Proxy来反射调用所有的接口方法   
  
 优劣：   
  
 CGLIB：会在内存中多存额外的class信息，对metaspace区的使用有影响，但是性能好，可以访问非接口的方法   
 JDK动态代理：本质是生成一个继承所有接口的Proxy来反射调用方法，局限性在于其只能代理接口的方法   
  
  
 引申点：   
  
 Spring的AOP实现以及应用场景   
 反射的开销：检查方法权限，序列化以及匹配入参   
 ASM   
  
  
 10.Java序列化的方式。   
 继承Serializable接口并添加SerializableId（idea有组件可以直接生成），ID实际上是一个版本，标志着序列化的结构是否相同   
 11.传值和传引用的区别，Java是怎么样的，有没有传值引用。   
 本质上来讲Java传递的是引用的副本，实际上就是值传递，但是这个值是引用的副本，比如方法A中传入了一个引用ref，那么在其中将ref指向其他对象并不影响在方法A外的ref，因为ref在传入方法A的时候实际上是指向同一个对象的另一个引用，可以称之为ref'，ref'若直接修改引用的对象会影响ref，但若ref'指向其他对象则和ref没有关系了   
 12.一个ArrayList在循环过程中删除，会不会出问题，为什么。   
 分情况讨论：   
  
 fori删除，不会直接抛异常，但是会产生异常访问   
  
  
 foreach删除（实际就是迭代器），会直接抛出并发修改异常，因为迭代器会进行获取迭代器时的exceptModCount和真实的modCount的对比   
  
  
 引申点：   
  
 迭代器实现   
 ArrayList内部细节   
  
  
 13.@transactional注解在什么情况下会失效，为什么。   
 方法A存在该注解，同时被方法B调用，外界调用的是Class.B的方法，因为内部实际上的this.a的调用方式没走代理类所以不会被切面切到   
 数据结构和算法   
 1.B+树   
 出度为m的一颗树，节点的子女在[M/2,M]之间   
 叶子节点存储全量信息   
 非叶子节点只充当索引进行叶子节点的路由（内存友好、局部性友好）   
 底层的叶子节点以链表的形式进行相连（范围查找友好）   
 2.快速排序，堆排序，插入排序（其实八大排序算法都应该了解   
 快排：核心是分治logn   
 堆排：基于二叉树nlogn   
 插入：暴力n2   
 3.一致性Hash算法，一致性Hash算法的应用   
 一致性hash，将整个hash的输出空间当成一个环，环中设立多个节点，每个节点有值，当对象的映射满足上个节点和这个节点中间值的时候它就落到这个节点当中来   
 应用：redis缓存，好处是平滑的数据迁移和快速的rebalance   
  
 引申点：   
  
 一致性hash热点怎么处理：虚拟节点   
 redis如何实现的：客户端寻址   
  
  
  
JVM  
   
  
  
1.JVM的内存结构。  
  
 程序计数器：计算读到第几行了，类似一个游标   
 方法栈：提供JVM方法执行的栈空间   
 本地方法栈：提供native方法执行的栈空间   
 堆：存对象用的，young分eden,s0,s1，分配比例大概是8:1:1，Old只有一个区   
  
方法区：1.8后为metaspace，存class信息，常量池（后迁移到堆中），编译出来的热点代码等  
  
  
   
  
 引申点：   
  
  
  
 heap什么时候发生溢出   
   
 stack什么时候发生溢出   
 方法区什么时候发生溢出   
 hotspot code的机制   
 流量黑洞如何产生的   
  
  
  
 2.JVM方法栈的工作过程，方法栈和本地方法栈有什么区别。   
   
  
 方法栈是JVM方法使用的，本地方法栈是native方法使用的，在hotspot其实是用一个   
 3.JVM的栈中引用如何和堆中的对象产生关联。   
 引用保存地址，直接可以查找到堆上对应地址的对象   
 4.可以了解一下逃逸分析技术。   
 方法中开出来的local变量如果在方法体外不存在的话则称之为无法逃逸   
  
 可以直接分配在栈上，随着栈弹出直接销毁，省GC开销   
 消除所有同步代码，因为本质上就是个单线程执行   
  
  
 引申点：   
 JVM编译优化：   
  
 逃逸分析   
 栈上分配   
 分层编译与预热   
 栈上替换   
 常量传播   
 方法内联 ...   
  
  
   
 5.GC的常见算法，CMS以及G1的垃圾回收过程，CMS的各个阶段哪两个是Stop the world的，CMS会不会产生碎片，G1的优势。   
 常见算法：   
  
 标记清楚：存在内存碎片，降低内存使用效率   
 标记整理：整理可分为复制整理和原地整理，不存在内存碎片，但是需要额外的cpu算力来进行整理，若为复制算法还需要额外的内存空间   
  
 CMS流程：   
  
 初始标记(stw)：获得老年代中跟GCRoot以及新生代关联的对象，将其标记为root   
 并发标记：将root标记的对象所关联的对象进行标记   
 重标记：在并发标记阶段，并没有stw，所以会有一些脏对象产生，即标记完毕之后又产生关联对象修改   
 最终标记(stw)：最终确定所有没有脏对象的存活对象   
 并发清理：并发的清理所有死亡对象   
 Reset：重设程序为下一次FGC做准备   
  
 CMS优劣：   
  
 优点：   
  
 不像PN以及Serial一样全程需要stw，只需要在两个标记阶段stw即可   
 并发标记、清楚来提升效率，减少stw的时间和整体gc时间   
 在最终标记前通过预设次数的重标记来清理脏页减少stw时间   
   
 缺点：   
  
 仍然存在stw   
 基于标记清楚算法的GC，节省算力但是会产生内存碎片   
 并发标记和清楚会造成cpu的高负担   
   
  
 G1流程：   
 这个我只懂个大概，如下   
 分块分代回收，可分为youngGC和MixedGC，特点是可预测的GC时间（即所谓的软实时特性）   
  
 引申点：   
  
 是否进行过线上分析   
 GC日志是否读过，里面有什么信息   
 你们应用的YGC和FGC频率以及时间是多少   
 你清楚当前应用YGC最多的一般是什么吗   
  
  
  
 业务相关：   
  
 在线上大部分curd业务当中，实际上造成ygc影响较严重且可优化的是日志系统   
 对dump出来的堆进行分析的话里面有很大一块是String，而其中大概率会是日志中的各种入参出参   
 优化方案有很多：   
  
 将不需要打日志的地方去除全量日志打印功能   
 日志在不同环境分级打印   
 只打出错误状态的日志   
 在大促期间关闭非主要日志打印   
 同步改异步等   
   
  
  
 6.标记清除和标记整理算法的理解以及优缺点。   
 上文已答   
 7.eden survivor区的比例，为什么是这个比例，eden survivor的工作过程。   
 8:2   
 定性的来讲：大部分对象都只有极短的存活时间，基本就是函数run到尾就释放了，所以给新晋对象的buffer需要占较多的比例，而s区可以相对小一点来容纳长时间存活的对象，较小的另一个原因是在几次年龄增长后对象会进入老年代   
 定量的来讲：实验所得，也可以根据自己服务器的情况动态调整（不过笔者没调过）   
 8.JVM如何判断一个对象是否该被GC，可以视为root的都有哪几种类型。   
 没有被GCRoot所关联   
 Root对象：（tips：不用硬记，针对着JVM内存区域来理解即可）   
  
 函数栈上的引用：包括虚拟机栈和native栈   
 static类的引用：存在方法区内   
 常量池中的常量：堆中   
  
  
 引申点：   
  
 gc roots和ref count的区别   
  
  
 9.强软弱虚引用的区别以及GC对他们执行怎样的操作。   
 强：代码中正常的引用，存在即不会被回收   
 软：在内存不足的时候会对其进行GC，可用于缓存场景（类似redis淘汰）   
 弱：当一个对象只有弱引用关联的时候会被下一次GC给回收   
 虚：又称幽灵引用，基本没啥用，在GC的时候会感知到   
  
 引申点：   
  
 每个引用的使用场景   
  
  
 是否在源码或者项目中看到过or使用过这几种引用类型（ThreadLocal里用了WeakReference）   
  
  
   
 10.Java是否可以GC直接内存。   
 在GC过程中如果发现堆外内存的Ref   
 11.Java类加载的过程。   
 加载：从某个地方读取class数据   
 链接：   
  
 验证：检验class是否有效   
 准备：开辟class信息存放的空间以及常量初始化   
 解析：符号引用转直接引用   
  
 初始化：在真正init的时候为其在堆上分配内存   
 12.双亲委派模型的过程以及优势。   
 System -> Ext -> Bootstrap类加载器依次从子类到父类进行双亲委派   
 本质是默认的类加载器都会直接调用super来尝试进行加载   
 优势：在不手动介入类加载过程的情况下可以保证基本的类都由统一的类加载器进行load，在内存中统一   
 13.常用的JVM调优参数。   
 虚拟机相关：影响整个虚拟机   
 XMX:最大堆空间   
 XMS:最小堆空间   
 堆分区相关：动态调整适配机器   
 XX:NewRatio 新生代的比例   
 XX:SurvivorRatio S1S0和edun区的比例   
 GC机制相关：动态调整适配是否需要低延迟还是高吞吐   
 XX:CMSInitiatingOccupancyFraction=70   
 14.dump文件的分析。   
 jmap dump出来看看哪些文件比较大or比较多，然后找认为可能有问题的看看具体是哪个class，基本就能定位到代码进行排查了，需要辅以日志/全链路监控工具/arthas等工具进行使用   
 15.Java有没有主动触发GC的方式（没有）。   
 System.gc可以hint虚拟机进行GC，但是并不一定会执行   
 多线程   
 1.Java实现多线程有哪几种方式。   
  
 重写Thread   
 重写Runnable，赛到Thread里面   
  
 普通重写   
 匿名类重写   
 lambda重写   
   
 Callable+FutureTask   
  
 2.Callable和Future的了解。   
 Callable vs. Runnable   
 Callable可以返回结果，Runnable无法返回结果   
 Future就是Callable返回的结果，底层是用一个volatile的变量标志是否已经结束来让调用者知道任务执行状况   
 线程池+SynList+Future可以获取一组任务的执行情况   
 3.线程池的参数有哪些，在线程池创建一个线程的过程。   
 核心线程数：决定正常情况下最多有几个线程在执行任务   
 最大线程数：在队列将要溢出的时候最多能多开几个线程   
 缓冲队列：作为一层buffer提供多任务的缓冲   
 线程生成工厂：提供名字设置、打日志等功能   
 拒绝策略：1. 抛出异常丢弃任务 2. 不抛出异常直接拒绝任务 3. 丢弃前面的任务执行新任务 4. 由发起线程执行该任务   
 存活时间：无任务的线程存活时间   
 创建过程：判断核心线程数是否达标，未达标直接创建线程执行，达上限后塞入队列中，队列满了创建新线程直到最大线程数，然后执行拒绝策略   
 4.volitile关键字的作用，原理。   
 保证该变量在所有线程中的可见性   
 原理：1. 保证内存可见性 2. 禁止指令重排序（内存屏障，读前读屏障，写后写屏障）   
 5.synchronized关键字的用法，优缺点。   
 锁对象：直接锁住对象   
 锁方法：锁住方法对应的调用对象，若是静态方法则锁class   
 优点：使用简单，可重入   
 缺点：缺乏高级特性（condition\cutdownLatch等），1.6前性能不好   
 6.Lock接口有哪些实现类，使用场景是什么。   
 ReenterLock：可重入锁，类似syn，AQS实现，通过双端队列+volatile的count值，实现并发，线程抢占分exclusive和share   
 ReadLock：读锁，支持并发读   
 WriteLock：写锁，与所有读锁写锁互斥   
 CopyOnWrite：写的时候Copy出一份对象进行插入，然后原对象供读，在新对象准备完毕之后将引用指过去（这一步需要同步）   
 Segment：分段锁，1.8前concurrentHashMap实现   
 7.可重入锁的用处及实现原理，写时复制的过程，读写锁，分段锁（ConcurrentHashMap中的segment）   
 见上文   
 8.悲观锁，乐观锁，优缺点，CAS有什么缺陷，该如何解决。   
 悲观锁：认为锁是强竞争的，修改数据前先将数据锁定,禁止其他用户/线程对其修改   
 乐观锁：认为锁是弱竞争的，不对数据进行锁定,在写入数据的时候判断是否符合标准,符合标准则写入,不符合则返回错误信息,常见的乐观锁有CAS   
 CAS缺点：ABA问题,数据从A变成B再变成A,其中经历了两次逻辑处理,但是如果某个线程在变换完毕过后读取这个值认为它依然是初始值A来进行逻辑操作就可能出现问题   
 ABA问题解法：加版本号   
  
 引申点   
  
 MySQL锁类型   
 MVCC流程   
  
  
 9.ABC三个线程如何保证顺序执行。   
 解法1:   
 CountDownLatch, 主线程设置一个latch, 值为1, 启动A线程, 执行完毕再继续往下, B线程类似   
 解法2:   
 Condition, 设置两个condition, A执行完释放B的condition, B执行完释放C的condition   
 解法3:   
 FutureTask, A线程提交一个FutureTask, 然后在主线程阻塞等待返回结果再进行B线程   
 解法4:   
 volatile, A执行完设置volatile为1, Bwhile读取volatile为1时进行逻辑操作, 执行完设为2, Cwhile读取2   
 10.线程的状态都有哪些。   
 New: 刚创建还没开始运行   
 Runnable: 交由操作系统执行, 但是不一定在吃cpu时间片   
 Wait: 等待被notify   
 TimeWait: 具有超时的wait   
 Blocked: 对于某个资源产生争用陷入阻塞   
  
 引申点:   
  
 和操作系统线程状态的区别: 例如在OS里等待IO的线程在Java线程体系中的状态是什么   
  
  
 11.sleep和wait的区别。   
 sleep阻塞线程到点了就自己醒了   
 wait阻塞线程并释放当前对象的锁, 需要notify   
  
 引申点:   
  
 锁升级   
 wait等待的是什么   
 只有一个线程的时候它能wait吗?   
  
 当前线程必须拥有此对象的monitor（即锁），才能调用某个对象的wait()方法能让当前线程阻塞， （这种阻塞是通过提前释放synchronized锁，重新去请求锁导致的阻塞，这种请求必须有其他线程通过notify()或者notifyAll（）唤醒重新竞争获得锁）   
   
  
  
 12.notify和notifyall的区别。   
 Notify随机挑一个, 剩下的还在wait状态   
 NotifyAll唤醒全部一起争用, 大部分会处于blocked状态   
 13.ThreadLocal的了解，实现原理。   
 一个map, 里面维护了当前线程作为k, 自定义数据解构作为v的键值对, 可以获取当前线程的上下文   
 为了防止内存泄漏里面的entry用的是弱引用, 当外界没有任何引用指到threadLocal里面变量的时候会被清楚, 即线程消亡的时候其threadlocal里的值会在下一次被GC   
 数据库相关   
 1.常见的数据库优化手段   
  
 log同步刷盘改异步刷盘   
 集群的话强双写改异步同步   
 针对sql优化（explain慢sql）   
 添加索引   
  
 2.索引的优缺点，什么字段上建立索引   
 优点：查的快，支持range   
 缺点：大部分查询实际需要回表，索引建立会额外消耗内存和磁盘，对开发者的sql也有要求   
 字段：区分度大的字段   
 3.数据库连接池。   
 mybatis有自带的, 市面常用的一般是durid   
 4.durid的常用配置。   
 连接池数量,idletime,keepLive原则,是否autocommit,建立链接前是否握手等(数据库的基本配置都大差不差,其实我也没看过duird)   
 计算机网络   
 1.TCP，UDP区别。   
 TCP:面向链接\可靠交付\拥塞控制\线程到线程   
 UDP:面向报文\尽力交付\n:m传播   
 2.三次握手，四次挥手，为什么要四次挥手。   
 三次握手:   
  
 client send : SYN = 1, seq = x   
 server feedback : ACK = 1, SYN = 1, seq = y, ack = x+1   
 client feedback : ACK = 1, seq = x+1, ack = y+1   
  
 四次挥手:   
  
 client send : FIN = 1, seq = x, ack = z   
 server feedback : ACK = 1, ack = x+1   
 server sned : FIN = 1, seq = y, ack =q   
 client feedback : ACK = 1, ack = q+1   
  
 为什么四次挥手 : 全双工通道关闭需要双方通信   
 3.长连接和短连接。   
 HTTP1.0属于经典的短链接, 每次通信需要重新开tcp端口   
 Neety可以制造长链接, websocket也可以, 通过心跳保持链接稳定然后进行传输, 接受端口和握手开销   
 4.连接池适合长连接还是短连接。   
 取决于连接池如何使用   
 如果是数据库连接池的话可以考虑使用长连接, 因为链接目标是一定的, 可以减小重复链接的开销   
 但是如果是RPC调用的话长短链接都可以   
 短链接的优势在于在少量请求随机请求到海量服务时不需要维持额外的开销去保持链接   
 长链接的优势在于如果调用服务比较固定, 那么长连接可以减少握手开销, 自动探活   
 设计模式   
 1.观察者模式   
 举例子wait/notify, 在观察到变化的时候就进行改变   
 2.代理模式   
 举例子JDK动态代理，通过一层proxy对真实对象进行代理，进行一些额外操作（e.g.:增强行为、负载均衡等）   
 3.单例模式，有五种写法，可以参考文章单例模式的五种实现方式   
  
 普通单例   
 lazyloading+syn单例   
 lazyloading+doublecheck单例   
 枚举   
 最后一种不知道，查了发现是静态内部类单例，利用静态内部类第一次访问才加载的机制实现lazyloading   
  
 4.可以考Spring中使用了哪些设计模式   
 工厂/单例/适配器/代理等   
 分布式相关   
 1.分布式事务的控制。   
 XA -> 2PC -> 3PC   
 XA:引入单点协调器   
 2PC:二阶段提交, prepare+commit, 但是问题在于commit阶段不知道能不能成功, 所以一旦超时就只能默认失败   
 3PC:三阶段提交, 和2PC区别就是commit的拆分为两个阶段, 先让所有执行者执行但是不commit, 然后统一commit, 可以提高成功率, 因为语句已经执行完毕了只差commit   
 2.分布式锁如何设计。   
 考虑设计要素   
  
 过期时间设置   
 是否需要续约   
 key是什么(前缀+业务key+线程uuid)   
 如何让其可重入(鉴权+续约)   
 如何防止ABA问题(线程A锁了之后, 超时释放, B又锁了一个, A错误释放)   
 如何原子释放(lua脚本走cas)   
  
 3.分布式session如何设计。   
 考虑设计要素:   
  
 过期时间设置   
 单点登录实现   
 续约设置   
 放脱裤存信息设置   
  
 4.dubbo的组件有哪些，各有什么作用。   
 duboo不熟悉   
 5.zookeeper的负载均衡算法有哪些。   
 zookeeper就会个zab，不过负载均衡无非是公平轮询、加权轮询、随机轮询或者维护某些资源信息的动态路由这几种   
 6.dubbo是如何利用接口就可以通信的。   
 不太熟，估计涉及到服务注册以及序列化反序列化相关内容   
 缓存相关   
 1.redis和memcached的区别。   
 memcached不熟, 不瞎吹牛B了   
 2.redis支持哪些数据结构。   
 String: SDS支持, 支持常数时间获取长度, 防缓冲区溢出   
 Set: 无序集合   
 Zset: 带score的无序集合, 跳表支持   
 List: 字符串列表，按照插入顺序排序, 双向链表支持   
 Hash: 是一个map, 可以存储解构性数据, rehash支持类似copyonwrite的感觉, 渐进式hash   
 3.redis是单线程的么，所有的工作都是单线程么。   
 严格意义上来讲redis的网络IO是单线程的, 但是并不是所有的工作都是单线程的   
 IO事件: 多路复用程序监听多个socket, 然后交给事件分发器有序的交到各个handler中进行处理   
 时间事件: 处理过期键\处理持久化\定时任务等   
 4.redis如何存储一个String的。   
  
 存储一个len   
 超长会扩容, 扩容会留一定的buffer   
 支持所有二进制存储, 不以/0为判断标准   
  
 5.redis的部署方式，主从，集群。   
 主从: master/slave, slave同步所有写事件   
 Sentinel: 主从模式下主挂了可以通过sentinel进行选主(CP, 会存在一段时间不可用)   
 集群: 16384slot, 每个节点需要分配一段的slot进行处理, 当所有slot都有节点在处理的时候才可以上线   
 6.redis的哨兵模式，一个key值如何在redis集群中找到存储在哪里。   
 sentinel模式下直接找master就行了   
 7.redis持久化策略。   
 AOF: 类似binlog, 对执行的写事件都写入log中, 复原的时候直接读取并执行就行了   
 RDB: 快照式持久化, 保存当前库内的全量key   
 框架相关   
 1.SpringMVC的Controller是如何将参数和前端传来的数据一一对应的。   
 mapping   
 2.Mybatis如何找到指定的Mapper的，如何完成查询的。   
 mybatis会读取xml文件, 并获取xml和interface的映射, 将需要执行的sql绑定在interface上, 并构造代理注入spring, 在调用时通过反射获取当前调用的interface以及method, 然后在注册好的映射map中获取具体执行的sql并执行   
 3.Quartz是如何完成定时任务的。   
 没研究过   
 4.自定义注解的实现。   
 反射+获取runtime期间的注解   
 5.Spring使用了哪些设计模式。 上文提到了   
 6.Spring的IOC有什么优势。   
 控制反转, 不需要在代码中手动的去控制对象的生灭与周期, 将生命周期交由Spring进行处理   
 7.Spring如何维护它拥有的bean。   
 一些较新的东西   
 1.JDK8的新特性，流的概念及优势，为什么有这种优势。   
  
 Interface可用default方法   
 Stream   
 Optional   
  
 流的概念: 一组不间断的数据流   
 优势: 处理数据的时候可以将一个集合当作集合来看待, 而不是一组对象的拼接, 对集合的操作方便很多   
 2.区块链了解   
 不太了解   
 3.如何设计双11交易总额面板，要做到高并发高可用。   
 binlog同步+join+导入OLAP/Search型存储中, 可以采用时间分片来降低计算成本

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第22篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

三年经验阿里Java社招面经分享  
  
精  
  
  
编辑于 2020-06-16 17:02:40  
  
 第一轮：   
 1、先自我介绍，包含日常工作   
 2、问了父子线程怎么共享数据 interitableThreadLocals   
 3、lock和sync区别   
 4、HashMap1.7、1.8区别   
 5、AQS原理（执行过程源码，入队出队的细节，源码细节）   
 6、CountDownLatch和CyclicBarrier的区别是什么源码级别   
 7、volatile从指令重排序，内存屏障，聊到总线风暴   
 8、mysql索引：聚集索引、非聚集索引、索引结构，顺带会问各种树的特性   
 9、举例优化sql   
 10、MVCC和事务隔离级别的关系   
 11、间隙锁、行锁、乐观锁悲观锁等   
 12、唯一索引和普通索引的区别   
 13、聊到了changeBuffer、页分裂合页合并   
 14、可达性分析算法中根节点有哪些   
 15、cms和G1区别   
 16、怎样GC调优   
 17、怎样排查CPU彪高、内存彪高   
 18、redis数据结构、跳跃表   
 19、redis qps能上多少，怎么知道的   
 20、sentinel和cluster区别和各自适用场景   
 21、redis cluster集群同步过程   
 22、redis单线程为什么快   
 23、多大叫大key、热key产生原因和后果以及怎么解决   
 24、本地缓存需要高时效性怎么办   
 25、spring的作用；   
 26、spring循环依赖怎么解决（说出三级缓存源码细节）；   
 27、spring aop原理（动态代理）、   
 28、spring bean生命周期（源码细节，以及各个位置的设计思路，有什么可扩展的）   
 29、dubbo服务暴露和引用过程，负载均衡策略，容错机制在哪里实现的源码   
 30、项目中碰到的问题。   
 31、为什么换工作，如果通过会直接说有笔试题，和你确认笔试时间。   
   
 第二轮   
 笔试两道题，第一题写代码，第二题写技术方案，以查询为主，考察锁粒度、时间粒度上的细节点。   
   
 第三轮   
 1、自我介绍、项目介绍   
 2、说到缓存穿透，让我设计一个防止缓存穿透的解决方案，简单的就是存null值，但肯定会深究，可以结合布隆过滤器，设计分布式系统，里面又会问到流量分发到具体过滤器服务的方式，比如一致性hash算法，怎么调用？比如dubbo直连、等等细节会边说边问   
 3、有没有做限流，设计一个侵入性最小的限流服务。   
 4、项目中碰到的问题，最好说框架本身问题，能提现个人能力，也避免问题太低级被面试官看low，刚好之前有发现一个dubbo的bug，所以这问题应该回答的还可以   
 5、为什么换工作   
   
 第四轮   
 1、介绍最熟悉的项目，业务上有没有什么优化点；和同行业其他公司的差距和优势   
 2、dubbo服务调用过程   
 3、NIO、BIO区别，NIO解决了什么问题，Netty线程模型（源码拷问）。   
 4、MQ相关   
   
 第五轮   
 1、项目介绍   
 2、听到说做了限流，限流标准（并发数？ qps？并发数和qps关系？说出了5种限流方案和对应算法原理）   
 3、dubbo调用端怎么在jvm中生成对应服务？dubbo服务端和调用端超时时间设置和区别、dubbo长连接。   
 4、mysql行锁最大并发数？（秒杀项目指出）   
 5、设计秒杀系统，我说的异步的方式，会问怎么优化？改为了同步的方式，异步和同步区别？   
 6、碰到哪些技术难点？怎么解决？有没有参考其他大厂？其他大厂方案什么样的？有没有关注阿里这边最新的技术？   
 7、刚刚的秒杀系统，会涉及到多个库表的更新，分布式事务怎么解决，我说的消息最终一致性，异步？有没有更好的方案？同步TCC方式，TCC方式原理？（三个阶段的具体实现）   
   
 第六轮   
 主管视频面试：个人介绍、项目介绍为主，十五分钟结束。   
   
 第七轮   
 HR面试：项目介绍、职位介绍、离职原因、当前薪资，如果没什么问题，一天后会电话反馈待遇并确认是否接受。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第23篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里社招新零售一二面  
  
编辑于 2020-06-10 15:59:30  
  
一面：  
  
 自我介绍  
 问项目  
 Java的多态  
 类的关系，组合和聚合的区别、哪个关系更紧密  
 线程池，线程池参数，线程间的通信  
 介绍ThreadLocal  
 ConcurrentHashMap，JDK1.7和1.8的不同实现  
 Hash算法  
 什么是时间复杂度  
 MySQL事务隔离级别  
 JVM内存分区  
 JVM指令和汇编指令  
  
 二面：  
  
 问项目，根据项目问问题  
 用过数据库索引吗？介绍如何使用，何时使用？  
 B树、B+树、红黑树  
 数据库分页查询，如何分页，怎么实现？  
 四种隔离级别，虚读和幻读什么意思发生在哪？  
 使用的MySQL版本，和之前版本的区别？  
 1.7和1.8的JVM有哪些不同？  
 垃圾回收算法，CMS垃圾回收器简单介绍  
 ConcurrentHashMap的size()函数1.7和1.8的不同，或者介绍一下如果是你如何设计？  
 排序算法，介绍一下快速排序，快速排序时间复杂度，是不是稳定排序，介绍几种你所知道的稳定排序算法  
 UDP，TCP，HTTP介绍一下，OSI七层模型  
 Redis支持的数据结构

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第24篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴Java工程师社招凉经  
  
编辑于 2020-06-08 10:11:44  
  
   
 一面：   
 1、先介绍一下自己吧   
 2、说一下自己的优缺点   
 3、具体讲一下之前做过的项目   
 4、你觉得项目里给里最大的挑战是什么？   
 5、Hashmap为什么不用平衡树？   
 6、AQS知道吗？知道哪一些呢？讲一讲。   
 7、CLH同步队列是怎么实现非公平和公平的？   
 8、ReetrantLock和synchronized的区别   
 9、讲一下JVM的内存结构   
 10、JVM 里 new 对象时，堆会发生抢占吗？你是怎么去设计JVM的堆的线程安全的？   
 11、讲一下redis的数据结构   
 12、redis缓存同步问题   
 13、讲一讲MySQL的索引结构   
 14、你有什么问题要问我吗？   
   
 二面：   
 1、根据项目问了一些细节问题   
 2、说一下HashMap的数据结构   
 3、红黑树和AVL树有什么区别？   
 4、如何才能得到一个线程安全的HashMap？   
 5、讲一下JVM常用垃圾回收期   
 6、redis分布式锁   
 7、再描述一下你之前的项目吧   
 8、你觉得这个项目的亮点在哪里呢？   
 9、你设计的数据库遵循的范式？   
 10、你有没有什么问题？   
   
 三面：   
 1、又聊项目   
 2、在项目中，并发量大的情况下，如何才能够保证数据的一致性？   
 3、elasticsearch为什么检索快，它的底层数据结构是怎么样的？   
 4、JVM内存模型   
 5、netty应用在哪些中间件和框架中呢？   
 6、线程池的参数   
 7、讲一下B树和B+树的区别   
 8、为什么要用redis做缓存？   
 9、了解Springboot吗？那讲一下Springboot的启动流程吧   
 10、如何解决bean的循环依赖问题？   
 11、Java有哪些队列？   
 12、讲一讲Spring和Springboot的区别   
 13、最近看了什么书？为什么？   
 14、你平时是怎么学习Java的呢？

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第25篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴社招面经分享-Java岗  
  
发布于 2020-06-07 19:56:00  
  
 一面:   
 1.进程和线程的区别；   
 2.什么叫线程安全？举例说明；   
 3.OSI七层模型，包括TCP,IP的一些基本知识；   
 4.数据库的锁；   
 5.DFS,BFS算法；   
 6.还有一些诸如collection framework的Java基础；   
 7、http中，get post的区别；   
 8、有个每秒钟5k个请求，查询手机号所属地的笔试题(记得不完整，没列出)，如何设计算法?请求再多，比如5w，如何设计整个系统？   
 9、比较熟悉什么技术？   
     
 二面：   
 1、是否愿意去杭州？   
 2、熟悉各种技术框架么？源码读过么？   
 3、数据库了解什么?mysql了解么？   
 4、前台技术，看我用的ExtJS，就问是否了解JQuery；   
 5、进现在的公司，觉得对技术是否有提升？   
 6、是否了解augularJS， nodejs，cmd规范，amd规范等；   
 7、jvm性能调优都做了什么；   
 8、高并发情况下，我们系统是如何支撑大量的请求的？   
 9、集群如何同步会话状态；   
 10、负载均衡的原理；   
 11、数据库事务属性；   
 12、二叉树的遍历算法；   
 13、hashtable和hashmap的区别；   
 14、并发、同步的接口或方法；   
 15、string、stringbuilder、stringbuffer区别；   
 16、https处理的一个过程，对称加密和非对称加密；   
 17、线程的几种状态；   
     
 三面：   
 1、struts1.2和webwork的区别   
 2、hibernate和ibatis的区别   
 3、spring工作机制，IOC容器   
 4、servlet的一些相关问题   
 5、webservice相关   
 6、java基础：jvm，HashSet等等   
 7、考察学习新技术的能力   
 8、前后端优化的基本常识，比如js放在后面不阻塞等；原生js的理解，正则，时间冒泡等。   
 9、firebug，fiddle等调试工具；如果是基本使用，看下ext框架，jquery等熟悉程度。jquery插件机制，sizzle选择器。   
 10、作用域链， 闭包的理解。   
     
 HR面：   
 1、愿意去杭州发展么？我说要考虑，hr就问主要考虑什么问题？   
 2、从技术角度来说，你觉得你跟你同学比怎么样？   
 3、你的父母怎么看待你的；   
 4、个人优势；   
 5、与同事沟通的时候，如果遇到冲突了如何解决？   
 15、工作中觉得哪方面欠缺？   
 16、有问题要问么？   
 17、期望薪水；   
 18、为什么要离开现在的公司？

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第26篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里社招面经分享（Java工程师）  
  
编辑于 2020-06-06 17:53:24  
  
 1、ArrayList,HashMap等集合框架类，重要基础类String等的处理。HashMap的底层存储结构等。（也可以顺带考察下对接口，继承，重载重写的基本理解；以及异常处理的注意事项）。   
   
 2、常见的查找算法及时间复杂度。常见的排序算法及时间复杂度。   
 3、比较重要的数据结构，如链表，队列，栈的基本理解及大致实现。   
 4、多线程问题：   
 （1）线程安全问题。 HashMap 是否线程安全，为何不安全。 ConcurrentHashMap，线程安全，为何安全。底层实现是怎么样的。   
 （2）J.U.C下的常见类的使用。 ThreadPool的深入考察； BlockingQueue的使用。（take，poll的区别，put，offer的区别）；原子类的实现。   
 （3）各种常见锁使用；   
   
 5、Java内存分代模型，GC算法，JVM常见的启动参数； CMS算法的过程。   
 6、Linux使用与问题分析排查   
 （1）grep，awk，sed； 是否自己写过shell脚本；   
 （2）常见的cpu load过高，us过高，一般是什么问题。引申出是否用过top，jstat，jstack等。   
 （3）常见的内存问题一般有哪些。 引申出是否用过free，top， jmap等。   
   
 7、spring aop的底层实验原理。 aop与cglib，与asm的关系。   
 8、spriong ioc的生命周期，init-method，intilizingbean接口方法afterPropertiesSet的先后顺序；   
   
 9、Hibernate对一二级缓存的使用，Lazy-Load的理解；   
 10、mysql存储引擎中索引的实现机制；   
 11、数据库事务的几种粒度；   
 12、行锁，表锁；乐观锁，悲观锁   
 13、HTTPS协议，SSL协议及完整交互过程；   
 14、redis，memcache底层客户端使用一致性Hash，看是否了解；redis的事件驱动多路复用底层实现；引申到NIO编程， 看对Netty，或mina是否了解。如果候选者同时用过memcache，redis，看下是否了解两者在使用场景上的区别。以考察使用深度，以及是否有好奇精神。   
   
 15、设计模式与重构   
 常见设计模式：如singlen，factory，abstract factory，strategy，chain，adaptor，decorator，composite，template，absever等。   
   
 16、最近上过上哪些技术站点； 最近在看哪些书？   
 17、了解加班接受程度情况，了解是否愿意在一段时间996等；   
 18、了解家庭婚姻状况，了解购房及居住地，籍贯情况，了解目前薪资，评估稳定性；

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第27篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴Java社招面经分享  
  
编辑于 2020-06-05 15:47:29  
  
 一面：   
 1、hashmap、concurrenthashmap底层实现和区别；   
 2、spring框架的原理；   
 3、如何写一个orm框架？   
 4、hibernate一级缓存和二级缓存，hibernate其他缓存；   
 5、hibernate事务传播行为种类；   
 6、springmvc原理；   
 7、restful的好处；   
 8、restful有几种请求，表单如何提交put请求？   
 9、web中安全性问题的考虑，如何防止？   
 10、web系统整体架构；   
 11、hibernate如何实现声明式事务？   
 12、java并发包；   
 13、volatile；   
 14、平常都看哪些书？   
 15、spring底层数据结构？   
 16、如何进行反射，如何提高反射的性能？   
 17、如何实现java的代理，为什么需要实现接口？   
 18、TCP协议三次握手？   
 19、springmvc用过哪些注解？   
 20、springAOP可以使用哪些代理，有什么区别？   
 21、为什么要分三层？   
 22、mybatis与hibernate有什么区别，为什么不用hibernate直接写sql语句？   
   
 二面：   
 1、hashmap,hashtable区别。   
 2、是否了解过hashmap的源码？   
 3、jvm相关，主要是内存分配的过程，何时出发GC，用什么工具或者命令来进行监控？   
 4、类加载，类隔离机制。   
 5、spring的ioc，aop，事物，然后顺便提到了数据库的事物，两阶段提交，数据库锁的级别，MySQL数据库锁的特殊之处（页级锁）   
 6、java线程池里面的arrayblockingqueue，linkedblockingqueue的用途和区别。   
 7、设计模式相关，主要讲解用途如何保证数据库集群中ID的唯一性，假设每秒钟并发20万次。   
 8、设计十万并发级别的网站后台，如何计算使用的ecs数目；   
 9、memcached和redis的异同和用途。   
 10、nosql数据库相关，主要讲了mongodb   
 11、10G的整数中，取出最大的一个；   
 12、synchronized，volatile，可重入锁的用途和优缺点。   
 13、java nio相关核心的api基本了解和应用   
 14、一致性hash   
 15、在一个大系统中存在哪些单点失效的问题   
 16、hadoop优化相关   
   
 三面：   
 1、什么是restful web service？   
 2、并发集合包？   
 3、项目中使用了哪些数据结构？   
 4、TCP中断连接四次挥手？   
 5、MD5加盐；   
 6、https相关；   
 7、支付宝接口开发；   
 8、知道哪些锁，有什么区别？   
 9、依旧JDK源码；   
 10、项目遇到什么问题，怎么解决？   
 11、JDK1.8有什么新特性？   
 12、聊了一下职业发展、职业规划；

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第28篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴java岗社招面经  
  
编辑于 2020-06-05 13:54:40  
  
 一面：   
 1、首先自我介绍；   
 2、数据结构算法的基本问题，如排序算法，二叉树遍历，后序遍历非递归，图的最短路径问题；   
 3、对一个数组进行绝对值排序的算法；   
 4、java中hashmap的底层实现；   
 5、java中垃圾回收机制GC原理等；   
 6、介绍自己的项目，数据库中用到的数据结构数据模型，死锁的概念，如何避免死锁?   
 7、乐观锁和悲观锁?   
 8、一致性hash算法；   
 9、项目中业务对象的关联关系/关联方式，谈谈左外连接及如何实现多对多关系模型；   
 10、spring的IOC和AOP；   
 11、sping如何实现（保证）事务一致性完整性，spring中事务传播机制类型；   
 12、谈谈WEB的架构，项目中的用户场景使用场景；   
 13、对分布式架构的了解，如分布式存储 分布式缓存 分布式计算；   
 14、谈谈对阿里的哪些开源项目有所了解？   
 15、对淘宝技术是否有了解？   
 16、对其杭州信息平台事业部其主要业务是否了解？   
   
 二面：   
 1、首先仍然是先自我介绍；   
 2、对哪些技术最熟悉最擅长？我说的是java WEB框架那些，然后就开始问java了   
 3、接口和抽象类的区别 实现有哪些不同？   
 4、序列化如何实现？用代码描述。writeObject? objectWrite?ObjectOutputStream.writeObject(). ObjectInputStream.readObject()；   
 5、IO框架 集合框架的描述。如何判断文件是否存在？如何读取一个目录下面的所有文件和子目录？代码描述。   
 6、广度优先遍历目录下面的所有子目录和文件？代码实现？List和Map的不同？HashMap的实现？如何解决散列冲突？   
 7、多线程机制？线程休眠10秒？sleep()与wait()的区别。   
 8、缓冲流buffer的用途？原理？   
 9、linux下如何查看CPU负载和IO端口的使用情况？如何创建目录？如何搜索一个指定的字符？   
 10、考察数据库的sql操作，给定了一个具体的业务表，如何对不同类型的数据进行统计。   
 11、事务的概念？脏读？如何避免？如何在代码中实现？   
 12、sql代码中如何使用如何定义一个事务？应该是如何定义一个事务的隔离级别？   
 13、项目中遇到的问题？解决方式？   
 14、SSI框架？spring如何定义一个bean?代码描述。Bean的生命周期？Bean的scope？   
 15、对淘宝技术架构的了解？从哪里获取到的？自己今后的职业规划？技术发展方向？   
 16、最后问我有什么想问的？谈了下信息平台事业部的主要业务；

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*第29篇\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

阿里巴巴/京东/美团Java社招面筋  
  
编辑于 2020-05-30 16:41:39  
  
 阿里巴巴   
   
 1、开发中Java用了比较多的数据结构有哪些？   
 2、谈谈你对HashMap的理解，底层原理的基本实现，HashMap怎么解决碰撞问题的？   
 3、这些数据结构中是线程安全的吗？假如你回答HashMap是线程安全的，接着问你有没有线程安全的map，接下来问了conurren包。   
 4、对JVM熟不熟悉？简单说说类加载过程，里面执行的哪些操作？问了GC和内存管理，平时在tomect里面有没有进行过相关的配置？   
 5、然后问了http协议，get和post的基本区别，接着tcp/ip协议，三次握手，窗口滑动机制。   
 6、开发中用了那些数据库？回答mysql，储存引擎有哪些？然后问了我悲观锁和乐观锁问题使用场景、分布式集群实现的原理。   
 7、然后问了我springmvc和mybatis的工作原理，有没有看过底层源码？   
   
 京东   
   
 1、Dubbo超时重试；Dubbo超时时间设置；   
 2、如何保障请求执行顺序；   
 3、分布式事物与分布式锁；（扣款不要出现负数）   
 4、分布式session设置；   
 5、执行某操作，前50次成功，第51次失败a全部回滚b前50次提交第51次抛异常，ab场景分别如何设置Spring？（传播性）   
 6、Zookeeper有哪些用？   
 7、JVM内存模型；   
 8、数据库垂直和水平拆分；   
 9、MyBatis如何分页；如何设置缓存；MySQL分页；   
 10、熟悉IO么？与NIO的区别，阻塞与非阻塞的区别；   
 11、分布式session一致性；   
 12、分布式接口的幂等性设计；「不能重复扣款」   
   
 美团   
   
 1、最近做的比较熟悉的项目是哪个？画一下项目技术架构图；   
 2、JVM老年代和新生代的比例？   
 3、YGC和FGC发生的具体场景；   
 4、jstack，jmap，jutil分别的意义？如何线上排查JVM的相关问题？   
 5、线程池的构造类的方法的5个参数的具体意义？   
 6、单机上一个线程池正在处理服务如果忽然断电该怎么办？（正在处理和阻塞队列里的请求怎么处理）？   
 7、使用无界阻塞队列会出现什么问题？   
 8、接口如何处理重复请求？   
 9、具体处理方案是什么？   
 10、如何保证共享变量修改时的原子性？   
 11、设计一个对外服务的接口实现类，在1,2,3这三个主机（对应不同IP）上；12、实现负载均衡和顺序轮询机制；（考虑并发）