**《Java后端技术面试汇总》**

作 者： Java后端技术 微信公众号

完成日期： 20180611

|  |  |  |  |
| --- | --- | --- | --- |
| 版本号 | 修改情况 | 修改人 | 修改日期 |
| v1.0 |  |  |  |
| v2.0 |  |  |  |
| v3.0 |  |  |  |
| v4.0 |  |  |  |
| v5.0 |  |  |  |



# 一、Java面试通关秘籍第一套

## ****1、基础篇****

### ****1.1、Java基础****

* 面向对象的特征：继承、封装和多态
* final, finally, finalize 的区别
* Exception、Error、运行时异常与一般异常有何异同
* 请写出5种常见到的runtime exception
* int 和 Integer 有什么区别，Integer的值缓存范围
* 包装类，装箱和拆箱
* String、StringBuilder、StringBuffer
* 重载和重写的区别
* 抽象类和接口有什么区别
* 说说反射的用途及实现
* 说说自定义注解的场景及实现
* HTTP请求的GET与POST方式的区别
* Session与Cookie区别
* 列出自己常用的JDK包
* MVC设计思想
* equals与==的区别
* hashCode和equals方法的区别与联系
* 什么是Java序列化和反序列化，如何实现Java序列化？或者请解释Serializable 接口的作用
* Object类中常见的方法，为什么wait  notify会放在Object里边？
* Java的平台无关性如何体现出来的
* JDK和JRE的区别
* Java 8有哪些新特性

### ****1.2、Java常见集合****

* List 和 Set 区别
* Set和hashCode以及equals方法的联系
* List 和 Map 区别
* Arraylist 与 LinkedList 区别
* ArrayList 与 Vector 区别
* HashMap 和 Hashtable 的区别
* HashSet 和 HashMap 区别
* HashMap 和 ConcurrentHashMap 的区别
* HashMap 的工作原理及代码实现，什么时候用到红黑树
* 多线程情况下HashMap死循环的问题
* HashMap出现Hash DOS攻击的问题
* ConcurrentHashMap 的工作原理及代码实现，如何统计所有的元素个数
* 手写简单的HashMap
* 看过那些Java集合类的源码

**1.3、进程和线程**

* 线程和进程的概念、并行和并发的概念
* 创建线程的方式及实现
* 进程间通信的方式
* 说说 CountDownLatch、CyclicBarrier 原理和区别
* 说说 Semaphore 原理
* 说说 Exchanger 原理
* ThreadLocal 原理分析，ThreadLocal为什么会出现OOM，出现的深层次原理
* 讲讲线程池的实现原理
* 线程池的几种实现方式
* 线程的生命周期，状态是如何转移的
* 可参考：《[Java多线程编程核心技术](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247484881&idx=2&sn=b0ecf85cd7c9e543c84e7a9859c20a26&chksm=e9c5fc60deb27576a6a9c453dabc585f43d9f29fd8a8f37ed0e7cc2f012c86b23fbd21763a39&scene=21#wechat_redirect)》

### ****1.4、锁机制****

* 说说线程安全问题，什么是线程安全，如何保证线程安全
* 重入锁的概念，重入锁为什么可以防止死锁
* 产生死锁的四个条件（互斥、请求与保持、不剥夺、循环等待）
* 如何检查死锁（通过jConsole检查死锁）
* volatile 实现原理（禁止指令重排、刷新内存）
* synchronized 实现原理（对象监视器）
* synchronized 与 lock 的区别
* AQS同步队列
* CAS无锁的概念、乐观锁和悲观锁
* 常见的原子操作类
* 什么是ABA问题，出现ABA问题JDK是如何解决的
* 乐观锁的业务场景及实现方式
* Java 8并法包下常见的并发类
* 偏向锁、轻量级锁、重量级锁、自旋锁的概念
* 可参考：《[Java多线程编程核心技术](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247484881&idx=2&sn=b0ecf85cd7c9e543c84e7a9859c20a26&chksm=e9c5fc60deb27576a6a9c453dabc585f43d9f29fd8a8f37ed0e7cc2f012c86b23fbd21763a39&scene=21#wechat_redirect)》

### ****1.5、JVM****

* JVM运行时内存区域划分
* 内存溢出OOM和堆栈溢出SOE的示例及原因、如何排查与解决
* 如何判断对象是否可以回收或存活
* 常见的GC回收算法及其含义
* 常见的JVM性能监控和故障处理工具类：jps、jstat、jmap、jinfo、jconsole等
* JVM如何设置参数
* JVM性能调优
* 类加载器、双亲委派模型、一个类的生命周期、类是如何加载到JVM中的
* 类加载的过程：加载、验证、准备、解析、初始化
* 强引用、软引用、弱引用、虚引用
* Java内存模型JMM

### ****1.6、设计模式****

* 常见的设计模式
* 设计模式的的六大原则及其含义
* 常见的单例模式以及各种实现方式的优缺点，哪一种最好，手写常见的单利模式
* 设计模式在实际场景中的应用
* Spring中用到了哪些设计模式
* MyBatis中用到了哪些设计模式
* 你项目中有使用哪些设计模式
* 说说常用开源框架中设计模式使用分析
* 动态代理很重要！！！

**1.7、数据结构**

* 树（二叉查找树、平衡二叉树、红黑树、B树、B+树）
* 深度有限算法、广度优先算法
* 克鲁斯卡尔算法、普林母算法、迪克拉斯算法
* 什么是一致性Hash及其原理、Hash环问题
* 常见的排序算法和查找算法：快排、折半查找、堆排序等

### ****1.8、网络/IO基础****

* BIO、NIO、AIO的概念
* 什么是长连接和短连接
* Http1.0和2.0相比有什么区别，可参考《[Http 2.0](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247484611&idx=1&sn=66c875392eedff8150633ddcd5d83e7a&chksm=e9c5fd72deb274648a607b9bc39bac34adadd768577b77354f6dc85422691605e210b69eeb7b&scene=21" \l "wechat_redirect" \t "_blank)》
* Https的基本概念
* 三次握手和四次挥手、为什么挥手需要四次
* 从游览器中输入URL到页面加载的发生了什么？可参考《[从输入URL到页面加载发生了什么](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247483724&idx=1&sn=e58dd30d124971c795584e8673d6cc71&chksm=e9c5f8fddeb271ebebbb6c350ed1abc252f1f26b4f35c4ce36e10bde9659a37520feabed2290&scene=21" \l "wechat_redirect" \t "_blank)》

## **2、数据存储和消息队列**

### ****2.1、数据库****

* MySQL 索引使用的注意事项
* DDL、DML、DCL分别指什么
* explain命令
* left join，right join，inner join
* 数据库事物ACID（原子性、一致性、隔离性、持久性）
* 事物的隔离级别（读未提交、读以提交、可重复读、可序列化读）
* 脏读、幻读、不可重复读
* 数据库的几大范式
* 数据库常见的命令
* 说说分库与分表设计
* 分库与分表带来的分布式困境与应对之策（如何解决分布式下的分库分表，全局表？）
* 说说 SQL 优化之道
* MySQL遇到的死锁问题、如何排查与解决
* 存储引擎的 InnoDB与MyISAM区别，优缺点，使用场景
* 索引类别（B+树索引、全文索引、哈希索引）、索引的原理
* 什么是自适应哈希索引（AHI）
* 为什么要用 B+tree作为MySQL索引的数据结构
* 聚集索引与非聚集索引的区别
* 遇到过索引失效的情况没，什么时候可能会出现，如何解决
* limit 20000 加载很慢怎么解决
* 如何选择合适的分布式主键方案
* 选择合适的数据存储方案
* 常见的几种分布式ID的设计方案
* 常见的数据库优化方案，在你的项目中数据库如何进行优化的

### ****2.2、Redis****

* Redis 有哪些数据类型，可参考《[Redis常见的5种不同的数据类型详解](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247483987&idx=1&sn=5c5e4cd5bc73a7e6f84e5d6adfab0935&chksm=e9c5fbe2deb272f4b5b75bd2ac92bb27950452623ec83c0e1add7e30c773160421fab1571680&scene=21" \l "wechat_redirect" \t "_blank)》
* Redis 内部结构
* Redis 使用场景
* Redis 持久化机制，可参考《[使用快照和AOF将Redis数据持久化到硬盘中](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247483992&idx=1&sn=8f554bc490c4db1a78a30144f873e911&chksm=e9c5fbe9deb272fff47483c241e6d2a7aae99dc8f6fe9fee31f2dd214d0cf81b33d51f7a7dbe&scene=21" \l "wechat_redirect" \t "_blank)》
* Redis 集群方案与实现
* Redis 为什么是单线程的？
* 缓存雪崩、缓存穿透、缓存预热、缓存更新、缓存降级
* 使用缓存的合理性问题
* Redis常见的回收策略

### ****2.3、消息队列****

* 消息队列的使用场景
* 消息的重发补偿解决思路
* 消息的幂等性解决思路
* 消息的堆积解决思路
* 自己如何实现消息队列
* 如何保证消息的有序性

## ****3、开源框架和容器****

### ****3.1、SSM/Servlet****

* Servlet的生命周期
* 转发与重定向的区别
* BeanFactory 和 ApplicationContext 有什么区别
* Spring Bean 的生命周期
* Spring IOC 如何实现
* Spring中Bean的作用域，默认的是哪一个
* 说说 Spring AOP、Spring AOP 实现原理
* 动态代理（CGLib 与 JDK）、优缺点、性能对比、如何选择
* Spring 事务实现方式、事务的传播机制、默认的事务类别
* Spring 事务底层原理
* Spring事务失效（事务嵌套），JDK动态代理给Spring事务埋下的坑，可参考《[JDK动态代理给Spring事务埋下的坑！](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247484940&idx=1&sn=0a0a7198e96f57d610d3421b19573002&chksm=e9c5ffbddeb276ab64ff3b3efde003193902c69acda797fdc04124f6c2a786255d58817b5a5c&scene=21" \l "wechat_redirect" \t "_blank)》
* 如何自定义注解实现功能
* Spring MVC 运行流程
* Spring MVC 启动流程
* Spring 的单例实现原理
* Spring 框架中用到了哪些设计模式
* Spring 其他产品（Srping Boot、Spring Cloud、Spring Secuirity、Spring Data、Spring AMQP 等）
* 有没有用到Spring Boot，Spring Boot的认识、原理
* MyBatis的原理
* 可参考《[为什么会有Spring](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247484822&idx=1&sn=6fbee2a12b31b6102a18d3725671d41b&chksm=e9c5fc27deb275319641c3f30d168b85c7c196fd276d47efa35046b5dc54f5b77174c5bf8808&scene=21#wechat_redirect)》
* 可参考《[为什么会有Spring AOP](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247484827&idx=1&sn=b9d82f3fced6a875f8dfc22e5849b28e&chksm=e9c5fc2adeb2753c516ef8fc959c0c9dd84ccacaa40473b64bc58b5137c30562a0b45803ba8e&scene=21#wechat_redirect)》

### ****3.2、Netty****

* 为什么选择 Netty
* 说说业务中，Netty 的使用场景
* 原生的 NIO 在 JDK 1.7 版本存在 epoll bug
* 什么是TCP 粘包/拆包
* TCP粘包/拆包的解决办法
* Netty 线程模型
* 说说 Netty 的零拷贝
* Netty 内部执行流程
* Netty 重连实现

**3.3、Tomcat**

* Tomcat的基础架构（Server、Service、Connector、Container）
* Tomcat如何加载Servlet的
* Pipeline-Valve机制
* 可参考：《[四张图带你了解Tomcat系统架构](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247484905&idx=1&sn=6c8acd89476fadbc4cb9ccfda9c9c2e4&chksm=e9c5fc58deb2754e7519511bb0ed8dcbfa3fe29179663b53f3626643f8b9c82068d9b0464ee6&scene=21#wechat_redirect)！》

## ****4、分布式****

## ****4.1、Nginx****

* 请解释什么是C10K问题或者知道什么是C10K问题吗？
* Nginx简介，可参考《[Nginx简介](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247483994&idx=1&sn=b6591f62c7ea6b4adc5a5bf1bf4eac40&chksm=e9c5fbebdeb272fdd865a9c61a380f6b909fc988f99d00ce0aa8c3efca501644db46c40bd4f2&scene=21" \l "wechat_redirect" \t "_blank)》
* 正向代理和反向代理.
* Nginx几种常见的负载均衡策略
* Nginx服务器上的Master和Worker进程分别是什么
* 使用“反向代理服务器”的优点是什么?

### ****4.2、分布式其他****

* 谈谈业务中使用分布式的场景
* Session 分布式方案
* Session 分布式处理
* 分布式锁的应用场景、分布式锁的产生原因、基本概念
* 分布是锁的常见解决方案
* 分布式事务的常见解决方案
* 集群与负载均衡的算法与实现
* 说说分库与分表设计，可参考《[数据库分库分表策略的具体实现方案](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247483931&idx=1&sn=6eda41aa81c1243422a603205d2fad22&chksm=e9c5fbaadeb272bc92537803c14a6f55e1170b1a3b8f60160f66417800c0ace960dfe192717a&scene=21#wechat_redirect)》
* 分库与分表带来的分布式困境与应对之策

**4.3、Dubbo**

* 什么是Dubbo，可参考《[Dubbo入门](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247483791&idx=1&sn=49345f1a022734e81e9257f2b8d38a52&chksm=e9c5f83edeb2712805a77c1e1589e8f1d04bd17e55eeb2a45cabddb46d03615636908f058628&scene=21" \l "wechat_redirect" \t "_blank)》
* 什么是RPC、如何实现RPC、RPC 的实现原理，可参考《[基于HTTP的RPC实现](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247483900&idx=1&sn=c5ca198a66a701f81c2ab118fe7a734a&chksm=e9c5f84ddeb2715bc574e467cd6537ef81f223453e0989ffd136976b48dcc2d961a75be596de&scene=21" \l "wechat_redirect" \t "_blank)》
* Dubbo中的SPI是什么概念
* Dubbo的基本原理、执行流程

## **5、微服务**

**5.1、微服务**

* 前后端分离是如何做的？
* 微服务哪些框架
* Spring Could的常见组件有哪些？可参考《[Spring Cloud概述](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247484125&idx=1&sn=ddba9fba6ae900f5ef71a68f70afebe5&chksm=e9c5fb6cdeb2727a1e8d16a05adb7d9df0170e392ae579d1e40075fb488d680f0c061aa45327&scene=21" \l "wechat_redirect" \t "_blank)》
* 领域驱动有了解吗？什么是领域驱动模型？充血模型、贫血模型
* JWT有了解吗，什么是JWT，可参考《[前后端分离利器之JWT](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247485183&idx=1&sn=05dac824dbb534710dd99d6c895fbaf5&chksm=e9c5ff4edeb27658173c8b06ad6d1241d3b7822c734ddf6ac064d40e63cb0cb0a0c90804b9c7&scene=21" \l "wechat_redirect" \t "_blank)》
* 你怎么理解 RESTful
* 说说如何设计一个良好的 API
* 如何理解 RESTful API 的幂等性
* 如何保证接口的幂等性
* 说说 CAP 定理、BASE 理论
* 怎么考虑数据一致性问题
* 说说最终一致性的实现方案
* 微服务的优缺点，可参考《[微服务批判](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247485005&idx=1&sn=78a1d286c6a15a81ea5dcf6634a70b54&chksm=e9c5fffcdeb276ea3c766a6e5954685db0e89bea8ff0f47c9a3ff2c0c02991f791a5160287c4&scene=21#wechat_redirect)》
* 微服务与 SOA 的区别
* 如何拆分服务、水平分割、垂直分割
* 如何应对微服务的链式调用异常
* 如何快速追踪与定位问题
* 如何保证微服务的安全、认证

### ****5.2、安全问题****

* 如何防范常见的Web攻击、如何方式SQL注入
* 服务端通信安全攻防
* HTTPS原理剖析、降级攻击、HTTP与HTTPS的对比

### ****5.3、性能优化****

* 性能指标有哪些
* 如何发现性能瓶颈
* 性能调优的常见手段
* 说说你在项目中如何进行性能调优

## ****6、其他****

### ****6.1、设计能力****

* 说说你在项目中使用过的UML图
* 你如何考虑组件化、服务化、系统拆分
* 秒杀场景如何设计
* 可参考：《[秒杀系统的技术挑战、应对策略以及架构设计总结一二！](http://mp.weixin.qq.com/s?__biz=MzI1NDQ3MjQxNA==&mid=2247485294&idx=1&sn=681b3fc8833bc978344f66c8dd33ff32&chksm=e9c5fedfdeb277c96e03e8943bff709ad5354cbbe0c0d894cdb9d1e8cc8da92bf51f1e832e37&scene=21#wechat_redirect)》

### ****6.2、业务工程****

* 说说你的开发流程、如何进行自动化部署的
* 你和团队是如何沟通的
* 你如何进行代码评审
* 说说你对技术与业务的理解
* 说说你在项目中遇到感觉最难Bug，是如何解决的
* 介绍一下工作中的一个你认为最有价值的项目，以及在这个过程中的角色、解决的问题、你觉得你们项目还有哪些不足的地方

### ****6.3、软实力****

* 说说你的优缺点、亮点
* 说说你最近在看什么书、什么博客、在研究什么新技术、再看那些开源项目的源代码
* 说说你觉得最有意义的技术书籍
* 工作之余做什么事情、平时是如何学习的，怎样提升自己的能力
* 说说个人发展方向方面的思考
* 说说你认为的服务端开发工程师应该具备哪些能力
* 说说你认为的架构师是什么样的，架构师主要做什么
* 如何看待加班的问题

# 二、Java面试通关秘籍第二套

## **1、Java相关**

* Arraylist与LinkedList默认空间是多少；
* Arraylist与LinkedList区别与各自的优势List 和 Map 区别；
* 谈谈HashMap，哈希表解决hash冲突的方法；
* 为什么要重写hashcode()和equals()以及他们之间的区别与关系；
* Object的hashcode()是怎么计算的？
* 若hashcode方法永远返回1或者一个常量会产生什么结果？
* Java Collections和Arrays的sort方法默认的排序方法是什么；
* 引用计数法与GC Root可达性分析法区别；
* 浅拷贝和深拷贝的区别；
* String s="abc"和String s=new String("abc")区别；
* HashSet方法里面的hashcode存在哪，如果重写equals不重写hashcode会怎么样？
* 反射的作用与实现原理；
* Java中的回调机制；
* 模板方法模式；
* 开闭原则说一下；
* 发布/订阅使用场景；
* KMP算法（一种改进的字符串匹配算法）；
* JMM里边的原子性、可见性、有序性是如何体现出来的，JMM中内存屏障是什么意思，

## **2、多线程**

* AtomicInteger底层实现原理；
* synchronized与ReentraLock哪个是公平锁；
* CAS机制会出现什么问题；
* 用过并发包下边的哪些类；
* 一个线程连着调用start两次会出现什么情况？
* wait方法能不能被重写，wait能不能被中断；
* 线程池的实现？四种线程池？重要参数及原理？任务拒接策略有哪几种？
* 线程状态以及API怎么操作会发生这种转换；
* 常用的避免死锁方法；

## **3、JVM**

* Minor GC与Full GC分别在什么时候发生？什么时候触发Full GC;
* GC收集器有哪些？CMS收集器与G1收集器的特点。
* Java在什么时候会出现内存泄漏；
* Java中的大对象如何进行存储；
* rt.jar被什么类加载器加载，什么时间加载；
* 自己写的类被什么加载，什么时间加载；
* 自己写的两个不同的类是被同一个类加载器加载的吗？为什么？
* 为什么新生代内存需要有两个Survivor区？
* 几种常用的内存调试工具：jmap、jstack、jconsole；
* 类加载的五个过程：加载、验证、准备、解析、初始化；
* G1停顿吗，CMS回收步骤，CMS为什么会停顿，停顿时间；
* 栈主要存的数据是什么，堆呢？
* 堆分为哪几块，比如说新生代老生代，那么新生代又分为什么？
* 软引用和弱引用的使用场景（软引用可以实现缓存，弱引用可以用来在回调函数中防止内存泄露）；

## **4、数据库**

* 数据库索引，什么是全文索引，全文索引中的倒排索引是什么原理
* 数据库最佳左前缀原则是什么？
* 数据库的三大范式；
* 悲观锁和乐观锁的原理和应用场景；
* 左连接、右连接、内连接、外连接、交叉连接、笛卡儿积等；
* 一般情况下数据库宕机了如何进行恢复（什么是Write Ahead Log机制，什么是Double Write机制，什么是Check Point）；
* 什么是redo日志、什么是undo日志；
* 数据库中的隔离性是怎样实现的；原子性、一致性、持久性又是如何实现的；
* 什么是组合索引，组合索引什么时候会失效；
* 关系型数据库和非关系型数据库区别；
* 数据库死锁如何解决；
* MySQL并发情况下怎么解决（通过事务、隔离级别、锁）；
* MySQL中的MVCC机制是什么意思，根据具体场景，MVCC是否有问题；
* MySQL数据库的隔离级别，以及如何解决幻读；

## **5、缓存服务器**

* Redis中zSet跳跃表问题；
* Redis的set的应用场合？
* Redis高级特性了解吗？
* Redis的pipeline有什么用处？
* Redis集群宕机如何处理，怎么样进行数据的迁移；
* Redis的集群方案；
* Redis原子操作怎么用比较好；
* Redis过期策略是怎么实现的呢？

## **6、SSM相关**

* Spring中@Autowired和@Resource注解的区别？
* Spring声明一个 bean 如何对其进行个性化定制；
* MyBatis有什么优势；
* MyBatis如何做事务管理；

## **7、操作系统**

* Linux静态链接和动态链接；
* 什么是IO多路复用模型（select、poll、epoll）；
* Linux中的grep管道用处？Linux的常用命令？
* 操作系统中虚拟地址、逻辑地址、线性地址、物理地址的概念及区别；
* 内存的页面置换算法；
* 内存的页面置换算法；
* 进程调度算法，操作系统是如何调度进程的；
* 父子进程、孤儿进程、僵死进程等概念；
* fork进程时的操作；
* kill用法，某个进程杀不掉的原因（僵死进程；进入内核态，忽略kill信号）；
* 系统管理命令（如查看内存使用、网络情况）；
* find命令、awk使用；
* Linux下排查某个死循环的线程；

## **8、网络相关**

* 数据链路层是做什么的?
* 数据链路层的流量控制？
* 网络模型的分层、IP和Mac地址在那个层、TCP和HTTP分别在那个层；
* TCP滑动窗口；
* TCP为什么可靠；
* TCP的同传，拆包与组装包是什么意思；
* Https和Http有什么区别；
* Http 为什么是无状态的；
* TCP三次握手，为什么不是三次，为什么不是四次；
* TCP的拥塞控制、流量控制详细说明？
* Http1.0和Http2.0的区别；
* 两个不同ip地址的计算机之间如何通信；
* 地址解析协议ARP；
* OSI七层模型分别对应着五层模型的哪一部分；
* TCP三次握手数据丢失了怎么办？那如果后面又找到了呢？

## **9、分布式相关**

* 消息队列使用的场景介绍和作用（应用耦合、异步消息、流量削锋等）；
* 如何解决消息队列丢失消息和重复消费问题；
* Kafka使用过吗，什么是幂等性？怎么保证一致性，持久化怎么做，分区partition的理解，LEO是什么意思，如何保证多个partition之间数据一致性的（ISR机制），为什么Kafka可以这么快（基于磁盘的顺序读写）；
* 异步队列怎么实现；
* 你项目的并发是多少？怎么解决高并发问题？单机情况下Tomcat的并发大概是多少，MySQL的并发大致是多少？
* 什么是C10K问题；
* 高并发情况下怎么办；
* 分布式理论，什么是CAP理论，什么是Base理论，什么是Paxos理论；
* 分布式协议的选举算法；
* 说一下你对微服务的理解，与SOA的区别；
* Dubbo的基本原理，RPC，支持哪些通信方式，服务的调用过程；
* Dubbo如果有一个服务挂掉了怎么办；
* 分布式事务，操作两个表不在一个库，如何保证一致性。
* 分布式系统中，每台机器如何产生一个唯一的随机值；
* 系统的量级、pv、uv等；
* 什么是Hash一致性算法？分布式缓存的一致性，服务器如何扩容（哈希环）；
* 正向代理、反向代理；
* 什么是客户端负载均衡策略、什么是服务器端负载均衡策略；
* 如何优化Tomcat，常见的优化方式有哪些；
* Nginx的Master和Worker，Nginx是如何处理请求的；

## **10、系统设计相关**

* 如何防止表单重复提交（Token令牌环等方式）；
* 有一个url白名单，需要使用正则表达式进行过滤，但是url量级很大，大概亿级，那么如何优化正则表达式？如何优化亿级的url匹配呢？
* 常见的Nginx负载均衡策略；已有两台Nginx服务器了，倘若这时候再增加一台服务器，采用什么负载均衡算法比较好？
* 扫描二维码登录的过程解析；
* 如何设计一个生成唯一UUID的算法？
* 实现一个负载均衡的算法，服务器资源分配为70%、20%、10%；
* 有三个线程T1 T2 T3，如何保证他们按顺序执行；
* 三个线程循环输出ABCABCABC....

## **11、安全相关**

* 什么是XSS攻击，XSS攻击的一般表现形式有哪些？如何防止XSS攻击；

# 三、Java面试通关秘籍第三套

## **1、基础题**

* 怎么解决Hash冲突；（开放地址法、链地址法、再哈希法、建立公共溢出区等）
* 写出一个必然会产生死锁的伪代码；
* Spring IoC涉及到的设计模式；（工厂模式、单利模式。。）
* toString()方法什么情况下需要重写；
* 判断对象相等时，什么情况下只需要重写 equals()，什么情况下需要重写 equals(),hashcode()？
* Set内存放的元素为什么不可以重复，内部是如何保证和实现的？
* 如何保证分布式缓存的一致性(分布式缓存一致性hash算法?)？分布式session实现？
* Java 8流式迭代的好处？
* 项目中用到的JDK的哪些特性？
* 说一下TreeMap的实现原理？红黑树的性质？红黑树遍历方式有哪些？如果key冲突如何解决？setColor()方法在什么时候用？什么时候会进行旋转和颜色转换？
* Spring的bean的创建时机？依赖注入的时机？
* ArrayList和LinkList的删除一个元素的时间复杂度；（ArrayList是O(N)，LinkList是O(1)）；
* CopyOnWriteArrayList是什么；
* 序列化和反序列化底层如何实现的（ObjectOutputStream 、ObjectInputStream、 readObject  writeObject）；
* 如何调试多线程的程序；
* 一个线程连着调用start两次会出现什么情况？（由于状态只有就绪、阻塞、执行，状态是无法由执行转化为执行的，所以会报不合法的状态！）
* HashMap在什么时候时间复杂度是O（1），什么时候是O（n），什么时候又是O（logn）；
* wait方法能不能被重写？（wait是final类型的，不可以被重写，不仅如此，notify和notifyall都是final类型的），wait能不能被中断；
* 一个Controller调用两个Service，这两Service又都分别调用两个Dao，问其中用到了几个数据库连接池的连接？

## **2、网络基础**

* HTTP、TCP、UDP的区别和联系；
* TCP和UDP各自的优势，知道哪些使用UDP协议的成功案例；
* TCP和UDP各用了底层什么协议；
* 单个UDP报文最大容量；
* 单个TCP报文最大容量；
* TCP报头格式、UDP报头格式；
* Server遭遇SYN Flood应当怎么处理；
* Web开发中如何防范XSS？
* 拆包和粘包的问题，如何解决，如果我们的包没有固定长度的话，我们的应用程序应该如何解决；

## **3、操作系统**

* 为什么要内存对齐；
* 为什么会有大端小端，htol这一类函数的作用；
* top显示出来的系统信息都是什么含义；（重要！）
* Linux地址空间，怎么样进行寻址的；
* Linux如何查找目录或者文件的；

## **4、分布式其他**

* 分库与分表带来的分布式困境与应对之策；
* Solr如何实现全天24小时索引更新；

## **5、Redis**

* Redis插槽的分配（key的有效部分使用CRC16算法计算出哈希值，再将哈希值对16384取余，得到插槽值）;
* Redis主从是怎么选取的（一种是主动切换，另一种是使用sentinel自动方式）;
* Redis复制的过程;
* Redis队列应用场景；
* Redis主节点宕机了怎么办，还有没有同步的数据怎么办;

## **6、系统设计开放性题目**

* 秒杀系统设计，超卖怎么搞;
* 你们的图片时怎么存储的，对应在数据库中时如何保存图片的信息的？
* 假如成都没有一座消防站，现在问你要建立几座消防站，每个消防站要配多少名消防官兵，多少辆消防车，请你拿出一个方案；
* 基于数组实现一个循环阻塞队列；
* 常见的ipv4地址的展现形式如“168.0.0.1”，请实现ip地址和int类型的相互转换。（使用位移的方式）
* 现网某个服务部署在多台Liunx服务器上，其中一台突然出现CPU 100%的情况，而其他服务器正常，请列举可能导致这种情况发生的原因？如果您遇到这样的情况，应如何定位？内存？CPU？发布？debug？请求量？

## **7、大数据量问题（后边会有专题单独讨论）**

* 给定a、b两个文件，各存放50亿个url，每个url各占64字节，内存限制是4G，让你找出a、b文件共同的url？
* 海量日志数据，提取出某日访问百度次数最多的那个IP；
* 一个文本文件，大约有一万行，每行一个词，要求统计出其中最频繁出现的前10个词，请给出思想，给出时间复杂度分析。

此话题后边会有专门的文章探讨，如果有等不及的小伙伴，可以移步参考：

1、https://blog.csdn.net/v\_july\_v/article/details/6279498

2、https://blog.csdn.net/v\_july\_v/article/details/7382693

## **8、逻辑思维题**

* 有两根粗细均匀的香（烧香拜佛的香），每一根烧完都花一个小时，怎么样能够得到15min？
* 假定你有8个撞球，其中有1个球比其他的球稍重,如果只能利用天平来断定哪一个球重,要找到较重的球,要称几次?（2次）；
* 实验室里有1000个一模一样的瓶子，但是其中的一瓶有毒。可以用实验室的小白鼠来测试哪一瓶是毒药。如果小白鼠喝掉毒药的话，会在一个星期的时候死去，其他瓶子里的药水没有任何副作用。请问最少用多少只小白鼠可以在一个星期以内查出哪瓶是毒药；（答案是10只）
* 假设有一个池塘，里面有无穷多的水。现有2个空水壶，容积分别为5升和6升。问题是如何只用这2个水壶从池塘里取得3升的水；

# 四、Java面试通关秘籍第四套

## **1、Java基础**

* 为什么JVM调优经常会将-Xms和-Xmx参数设置成一样；
* Java线程池的核心属性以及处理流程；
* Java内存模型，方法区存什么；
* CMS垃圾回收过程；
* Full GC次数太多了，如何优化；
* 直接内存如何管理的；
* Java线程池的几个参数的意义和实现机制；
* Java线程池使用无界任务队列和有界任务队列的优劣对比；
* CountDownLatch和CyclicBarrier的区别；
* Java中有哪些同步方案（重量级锁、显式锁、并发容器、并发同步器、CAS、volatile、AQS等）
* 如果你的项目出现了内存泄露，怎么监控这个问题呢；
* 标记清除和标记整理的区别和优缺点，为何标记整理会发生stop the world；
* 线程池，如何根据CPU的核数来设计线程大小，如果是计算机密集型的呢，如果是IO密集型的呢？
* 让你设计一个cache如何设计；
* String中hashcode是怎么实现的；
* JDK中哪些实现了单例模式？
* 多个线程同时读写，读线程的数量远远⼤于写线程，你认为应该如何解决并发的问题？你会选择加什么样的锁？
* 线程池内的线程如果全部忙，提交⼀个新的任务，会发⽣什么？队列全部塞满了之后，还是忙，再提交会发⽣什么？
* synchronized关键字锁住的是什么东西？在字节码中是怎么表示的？在内存中的对象上表现为什么？
* wait/notify/notifyAll⽅法需不需要被包含在synchronized块中？这是为什么？
* ExecutorService你一般是怎么⽤的？是每个Service放一个还是个项目放一个？有什么好处？

## **2、数据库**

* InnoDB的插入缓冲和两次写的概率和意义；
* 如果建了⼀个单列索引，查询的时候查出2列，会⽤到这个单列索引吗？（会用到）
* 如果建了⼀个包含多个列的索引，查询的时候只⽤了第⼀列，能不能⽤上这个索引？查三列呢？
* 接上题，如果where条件后⾯带有⼀个 i + 5 < 100 会使⽤到这个索引吗？
* like %aaa%会使⽤索引吗? like aaa%呢?
* drop、truncate、delete的区别？
* 平时你们是怎么监控数据库的? 慢SQL是怎么排查的？（慢查询日志）
* 你们数据库是否⽀持emoji表情，如果不⽀持，如何操作?选择什么编码方式？如果支持一个表情占几个字节?(utf8mb4)；
* 如果查询很慢，你会想到的第⼀个⽅式是什么？（数据库索引）

## **3、Linux基础**

* Linux下可以在/proc目录下可以查看CPU的核心数等；cat /proc/下边会有很多系统内核信息可供显示；
* 说一下栈的内存是怎么分配的；
* Linux各个目录有了解过吗？/etc、/bin、/dev、/lib、/sbin这些常见的目录主要作用是什么？
* 说一下栈帧的内存是怎么分配的；
* Linux下排查某个死循环的线程；
* 动态链接和静态链接的区别；
* 进程的内存分布；
* 如何查找一个进程打开所有的文件；
* 说一下常使用的协议及其对应的端口；
* 为什么会有内核态，保护模式你知道吗?
* 文件是怎么在磁盘上存储的？
* 有了进程为何还要线程呢，不同进程和线程他们之间有什么不同。（进程是资源管理的最小单位，线程是程序执行的最小单位。在操作系统设计上，从进程演化出线程，最主要的目的就是更好的支持SMP以及减小（进程/线程）上下文切换开销。）
* InnoDB聚集索引B+树叶子节点和磁盘什么顺序相同;
* 文件系统，进程管理和调度，内存管理机制、虚地址保护模式；

## **4、网络基础**

* HTTP1.0和HTTP1.1的区别；
* DHCP如何实现分配IP的； 发现阶段（DHCP客户端在网络中广播发送DHCP DISCOVER请求报文，发现DHCP服务器，请求IP地址租约）、提供阶段（DHCP服务器通过DHCP OFFER报文向DHCP客户端提供IP地址预分配）、选择阶段（DHCP客户端通过DHCP REQUEST报文确认选择第一个DHCP服务器为它提供IP地址自动分配服务）和确认阶段（被选择的DHCP服务器通过DHCP ACK报文把在DHCP OFFER报文中准备的IP地址租约给对应DHCP客户端）。
* OSI七层模型，每层都说下自己的理解和知道的，说的越多越好；

## **5、框架相关**

* Servlet如何保证单例模式,可不可以编程多例的哪？
* Dubbo请求流程以及原理；
* Spring框架如何实现事务的；
* 如果一个接⼝有2个不同的实现, 那么怎么来Autowire一个指定的实现？(可以使用Qualifier注解限定要注入的Bean，也可以使用Qualifier和Autowire注解指定要获取的bean，也可以使用Resource注解的name属性指定要获取的Bean)
* Spring框架中需要引用哪些jar包，以及这些jar包的用途；
* Spring Boot没有放到web容器⾥为什么能跑HTTP服务？
* Spring中循环注入是什么意思，可不可以解决，如何解决；
* Spring的声明式事务 @Transaction注解⼀般写在什么位置? 抛出了异常会⾃动回滚吗？有没有办法控制不触发回滚?
* MyBatis怎么防止SQL注入；
* Tomcat本身的参数你⼀般会怎么调整？
* 了解哪几种序列化协议？如何选择合适的序列化协议；
* Redis渐进式rehash过程？
* 比如我有个电商平台，做每日订单的异常检测，服务端代码应该写；

# 五、Java面试通关秘籍第五套

## **1、Java相关**

* 乐观悲观锁的设计，如何保证原子性，解决的问题；
* char和double的字节，以及在内存的分布是怎样；
* 对象内存布局，然后讲下对象的死亡过程？
* 对象头，详细讲下；
* sync原理详细，sync内抛异常会怎样，死锁吗？还是释放掉？怎么排查死锁？死锁会怎样？有没有什么更好的替代方案？
* 详细讲一下集合，HashSet源码，HashMap源码，如果要线程安全需要怎么做？
* 多线程是解决什么问题的？线程池解决什么问题？
* 线程池，如何设计的，里面的参数有多少种，里面的工作队列和线程队列是怎样的结构，如果给你，怎样设计线程池？
* AQS原理，ReentranLock源码，设计原理，整体过程。
* 继续聊多线程源码，sync原理，然后一个场景设计题；
* float f = 1.4f;double d = 1.4d; 与 float f = 1.5f;double d = 1.5d; 是否为true，内存是怎样的；
* split的源码，split("a|b|c");得出多少个数组；
* 把所有认识熟用的JUC( java.util.concurrent(简称JUC)包)下的类写出来，讲下使用，然后讲下原生的线程操作;
* 开闭原则，解析工厂方法模式，建造者模式，区别。手撸出来。
* 讲下JVM的大页模式，JVM内存模型;
* 什么是敏捷开发，防御性编程，并行编程。Team Leader的思考;
* 逃逸分析是什么，作用是什么，用途是什么;
* 怎么认为一个类是线程安全？线程安全的定义是什么？Java有多少个关键字进行同步？为什么这样设计？（聊了一大堆，一堆为什么）；
* 两个线程设计题。记得一个是：t1,t2,t3，让t1，t2执行完才执行t3，原生实现。
* 写个后缀表达式，为什么要设计后缀表达式，有什么好处？然后写下中缀。
* 我看你做过性能优化，比如你怎么分析项目里面的OOM的，内存泄露呢？详细说思路;
* 说下多线程，我们什么时候需要分析线程数，怎么分析，分析什么因素;
* 抽象方法和类方法的区别，static的抽象方法可以吗？
* 说下Java的克隆体系;
* 涉及OOM、JVM优化、源码问题、数据库优化、多线程等问题;
* CPU高？什么情况CPU高？解决什么问题？
* 你有遇到过临界区问题吗？有遇到过吗？你在项目遇到这个问题是怎样解决的？
* volatile关键字作用;
* Java的多态怎么实现;
* 解释一下自旋;
* 解释一下信号量;
* 什么情况下会触发类加载；
* Java内存抖动严重，优化的思路；

## **2、数据库相关**

* SQL优化思路，联合索引与底层树结构的映像关系，索引结构（B+、B-），为什么用这样的结构；
* 讲下MySQL的集群？集群遇到过什么问题？sql的优化？
* 你目前为止遇到的最大数据量是多少？知道100万时候怎么设计吗？1000万呢？过几十亿呢？
* MySQL有多少个参数可调，除了最大连接数。全部列出来，一个个分析。
* 聊下优化过的索引，怎么优化;
* 红黑树和平衡树的区别，为什么数据库不用红黑树;
* mysql有哪些锁，意向锁有什么用;
* 数据库高并发下的优化思路;
* 数据库什么情况下索引会失效;

## **3、数据结构和操作系统相关**

* 数据结构学过吧，聊一下？学过什么结构？讲下树和队列？B树呢？
* 操作系统学过吧，聊一下？讲一下系统内存是怎样的？分段分页虚拟内存？
* 页面置换算法呢？多少种？有最优的置换算法吗？
* 你学过什么课程？然后聊下操作系统，内核、用户之类。
* 反转链表手撸;
* 快排，给一串数组，把具体每次patition写下，最终结果也写45, 32, 41, 35, 38, 20, 50;
* 一个整数status, 判断第K个比特位是否为比特1;
* 把递归实现的快排改成非递归，你知道非递归有什么好处吗;
* 举例使用分治思想的算法;

## **4、网络相关**

* 讲下请求头细节？
* Http和Https？Http1.0,1.1,2.0，讲下长连接和短连接？Https是怎样的？如果我篡改了公钥呢？怎么防止？
* Get和Post，讲下区别，要我模拟出抓包来。
* 详细讲下Cookie和Session，Token，OAuth2.0协议;
* 拥塞算法知道吗？哪些，分别怎样？
* 学过计算机网络是吧？socket熟悉吗？对它的读写缓冲区有理解吗？怎么的？那滑动窗口是怎样的？为什么这样设计？
* 再聊下Http的Http basic authentication;
* Https的过程;

## **5、框架相关**

* 聊下Spring源码，知道多少，都聊一下；
* 聊下Spring注解，@Autowire，@Resource，以及他们的解析过程；
* 聊一下架构，接入层架构，服务层架构。聊下技术栈，Spring Boot，Spring Cloud、Docker；
* Spring ioc的具体优势，和直接New一个对象有什么区别;
* Servlet生命周期，是否单例，为什么是单例;
* Spring Mvc初始化过程；

## **6、分布式相关**

* 多少种RPC框架？
* 一致性哈希是干嘛的？
* 搭建高并发高可用系统需要怎样设计？考虑哪些东西，有多少说多少。
* 你对缓存有什么理解？缓存是解决什么问题？后端缓存有哪些，分别解决什么问题？
* 聊一下分布式锁；
* 你是怎么设计系统缓存的，为什么，什么场景；
* 也来说下，削峰的多种实现，Redis？MQ？
* 为什么用mq就能削峰？解决什么问题？

## **7、设计题**

* 有几台机器存储着几亿淘宝搜索日志，你只有一台2g的电脑，怎么选出搜索热度最高的十个搜索关键词;
* 如何设计算法压缩一段URL;
* 有一个页面能同时展示两个广告，现在有五个广告，设计算法使五个广告展示概率为1:2:3:4:5；
* 有25匹马，五个赛道，用最少比赛次数将25匹马排序；

## **8、其他相关**

* Tomcat缓存，聊下缓存的整体理解，知道多少种缓存；
* 解释下Mucene原理，倒排索引，怎样进行中文分词，基于什么进行分词；
* TopN的大数据量题；
* 你对接入层要思考什么东西？遇到过哪些问题？搭建系统要考量哪些因素？
* 然后项目问题，优化问题；
* 熟悉maven是吧？我们来聊下Maven的源码原理，Maven冲突的时候，怎么选择依赖包，我们怎么查，我们遇到两个不一样的版本，我们应该如何去选择，为什么？
* 项目如何分组，性能优化小组应该做哪些;
* 我们来说下接入层的搭建，认知分析;
* 问下项目的系统构建，思考，为什么这样构建？
* 如何判断一段代码的好坏;