



链滴

Java 高级工程师面试题及参考答案

作者: [wgl530](#)

原文链接: <https://ld246.com/article/1577972012408>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



Java 高级工程师面试题总结及参考答案

一、面试题基础总结

1、JVM 结构原理、GC 工作机制详解

答：具体参照：JVM 结构、GC 工作机制详解，说到 GC，记住两点：1、GC 是负责回收所有无引用对象的内存空间。注意：垃圾回收回收的是无任何引用的对象占据的内存空间而不是对象本身
2、GC 回收机制的两种算法，a、引用计数法 b、可达性分析算法（这里的可达性，大家可以看基础 2 Java 对象的什么周期），至于更详细的 GC 算法介绍，大家可以参考：Java GC 机制算法

2、Java 对象的生命周期

答：创建阶段、应用阶段、不可见阶段、不可达阶段、收集阶段、终结阶段、对象空间重新分配段等等，具体参照：Java 对象的生命周期

3、Map 或者 HashMap 的存储原理

答：HashMap 是由数组 + 链表的一个结构组成，具体参照：HashMap 的实现原理

4、当数据表中 A、B 字段做了组合索引，那么单独使用 A 或单独使用 B 会有索引效果吗？（使用 like 查询如何有索引效果）

答：看 A、B 两字段做组合索引的时候，谁在前面，谁在后面，如果 A 在前，那么单独使用 A 会有索引效果，单独使用 B 则没有，反之亦然。同理，使用 like 模糊查询时，如果只是使用前面 %，那么索引效果，如果使用双 % 号匹配，那么则无索引效果

5、数据库存储日期格式时，如何考虑时区转换问题？

答：使用 TimeStamp，原因参照：Java 编程中遇到的时区转换问题

6、Java Object 类中有哪些方法？

答：Object 有哪些方法

7、HTTP 协议，GET 和 POST 的区别

答：浅谈 HTTP 中 GET 和 POST 的区别

二、线程、设计模式、缓存方面

1、SimpleDateFormat 是非线程安全的，如何更好的使用而避免风险呢

答：关于 SimpleDateFormat 安全的时间格式化线程安全问题

2、如何看待设计模式，并简单说说你对观察者模式的理解

答：1、[设计模式](#)有神马用 2、观察者模式类图及实现

3、集群环境中，session 如何实现共享

答：

1、Java 集群之 session 共享

2、session 多服务器共享方案，还有一种方案就是使用一个固定的服务器专门保持 session，其他服务器共享

4、分布式、集群环境中，缓存如何刷新，如何保持同步？

答：

A、缓存如何刷新？ 1、定时刷新 2、主动刷新覆盖，每个缓存框架都有自带的刷新机制，或者说失效机制，就拿 Redis 和 Ehcache 举例，他们都有自带的过期机制，另外主动刷新覆盖时，只需取对应的 key 进行数据的覆盖即可

B、缓存如何保持同步？这个 Redis 有自带的集群同步机制，即复制功能，具体参考：[基于 Redis 分布式缓存实现](#)，Ehcache 也有分布式缓存同步的配置，只需要配置不同服务器地址即可，参照：[Ehcache 分布式缓存同步](#)

5、一条 SQL 执行过长的时间，你如何优化，从哪些方面？

答：

1、查看 SQL 是否涉及多表的联表或者子查询，如果有，看是否能进行业务拆分，相关字段冗余或者并成临时表（业务和算法的优化）

2、涉及链表的查询，是否能进行分表查询，单表查询之后的结果进行字段整合

3、如果以上两种都不能操作，非要链表查询，那么考虑对相对应的查询条件做索引。加快查询速度

4、针对数量大的表进行历史表分离（如交易流水表）

5、数据库主从分离，读写分离，降低读写针对同一表同时的压力，至于主从同步，MySQL 有自带的 innodb 实现主从同步

6、explain 分析 SQL 语句，查看执行计划，分析索引是否用上，分析扫描行数等等

7、查看 MySQL 执行日志，看看是否有其他方面的问题

个人理解：从根本上来说，查询慢是占用 MySQL 内存比较多，那么可以从这方面去酌手考虑

三、三大框架方面问题

1、Spring 事务的隔离性，并说说每个隔离性的区别

解答：Spring 事务详解

2、Spring 事务的传播行为，并说说每个传播行为的区别

解答：Spring 事务详解

3、hibernate 跟 Mybatis/ ibatis 的区别，为什么选择？

解答：Hibernate 与 MyBatis 的比较

4、Struts 跟 Spring mvc 的优缺点，让你选会如何选

解答：Spring MVC 与 Struts 的区别

5、简单说说 Spring 事务机制

解答：Spring 事务机制

6、Spring 4.0 新特性

解答：Spring4 新特性

四、负载均衡、集群相关

1、weblogic 负载均衡的原理和集群的配置

解答：a、WEBLOGIC 负载均衡原理 b、负载均衡和集群的配置（参考）

2、Nginx+Tomcat+Redis 实现负载均衡、资源分离、session 共享

解答：配置参考

3、nginx 配置文件详解——nginx.conf

解答：Nginx 配置文件详细说明

五、项目优化相关

1、Web 如何项目优化

解答：这个我整理过一次，Web 项目性能优化（整理）

2、单例模式有几种？如何优化？

解答：单例模式的 7 中用法

3、简单说说线程池的原理和实现

解答：线程原理及实现

六、并发和安全方面

1、**项目并发**如何处理？（我们是 Web 项目）

解答：高并发量网站解决方案，另外，还有数据库乐观锁，数据库读写分离、使用消息队列、多用存过程等等

2、简单说说功能权限存在的水平权限漏洞和垂直权限漏洞的场景和解决办法（因为我们目前权限级就是功能权限）

解答：

A、水平权限漏洞，如下图

假设机构有用户 A 和用户 B 两个用户，其中 A 有 1、2 和 3 权限，用户 B 有 2 和 3 的权限，这时候假设用户 B 知道 1，并给自己添加 1 的权限，这时候就是水平权限漏洞。

目前解决办法：1、限制入口，让用户 B 无法编辑自己的权限 2、对用户 B 无法进行向上扩展。最根本的解决办法是深入到数据权限

解答：水平权限漏洞和解决办法

B、垂直权限漏洞

解答：垂直权限漏洞案例和解决方案

3、平台上的图片如何防盗链

解答：http 下载防盗链原理：http 协议的字段 referer 记录来实现

4、如何区分上传的图片是不是木马？

解答：1、看上传的图片后缀 2、如何后缀是篡改的，那么每个文件有个魔术数字 文件上传-魔术数字

5、消息队列的原理和实现

解答：1、消息队列原理 2、深入浅出 消息队列 ActiveMQ

七、数据库方面

1、MySQL 查询字段区分不区分大小写？

解答：不区分，哪怕值也不区分（我当时还反问了，区分不区分大小的应用含义有哪些，面试官没说得来）

2、简单说说数据库集群和负载均衡、分布式（我不懂这块）

解答：数据库负载均衡和集群参考，参考 2

3、存储过程的结构和优点

解答：大概结构

存储过程的优缺点

4、触发器的原理和作用

解答：参考

八、Java 底层基础题

1、SpringMVC 的原理以及返回数据如何渲染到 jsp/html 上？

答：Spring MVC 的核心就是 DispatcherServlet，一个请求经过 DispatcherServlet，转发给 HandlerMapping，然后经反射，对应 Controller 及其里面方法的 @RequestMapping 地址，最后经 ModelAndView 和 ViewResolver 返回给对应视图。具体可参考：Spring MVC 的工作原理

2、一个类对象属性发生改变时，如何让调用者知道？

答：Java event 时间监听，即在 set 方法改变属性时，触发，这种模式也可以理解为观察者模式，体查看：观察者模式简单案例和说明

3、重写 equals 为何要重写 hashCode？

答：判断两个对象是否相等，比较的就是其 hashCode，如果你重载了 equals，比如说是基于对象的容实现的，而保留 hashCode 的实现不变，那么很可能某两个对象明明是“相等”，而 hashCode 不一样。hashCode 不一样，就无法认定两个对象相等了

4、谈谈你对 JVM 的理解？

答：Java 语言的一个非常重要的特点就是与平台的无关性。而使用 Java 虚拟机是实现这一特点的关键。Java 编译器只要面向 JVM，生成 JVM 能理解的代码或字节码文件。Java 源文件经编译成字节码序，通过 JVM 将每一条指令翻译成不同平台机器码，通过特定平台运行。

JVM 执行程序的过程：I.加载。class 文件，II.管理并分配内存，III.执行垃圾收集

JRE (Java 运行时环境) 由 JVM 构造的 Java 程序的运行环境

具体详情: JVM 原理和调优

5、MySQL 的事物隔离级别?

答: MySQL 的事物隔离级别 其实跟 Spring 的事物隔离级别一样, 都是 1、Read Uncommitted (取未提交内容), 2、Read Committed (读取提交内容), 3、Repeatable Read (可重读), 4、serializable (可串行化) 具体参照: MySQL 事物隔离级别

6、Spring 的原理

答: Spring 的核心是 IoC 和 AOP, IoC 是依赖注入和控制反转, 其注入方式可分为 set 注入、构造器注入、接口注入等等。IoC 就是一个容器, 负责实例化、定位、配置应用程序中的对象及建立这些对象间的依赖。简单理解就是: Java 每个业务逻辑处理至少需要两个或者以上的对象协作进行工作, 但每个对象在使用它的合作对象的时候, 都需要频繁的 new 对象来实现, 你就会发现, 对象间的耦合高了。而 IoC 的思想是: Spring 容器来管理这些, 对象只需要处理本身业务关系就好了。至于什么控制反转, 就是获得依赖对象的方式反转了。

AOP 呢, 面向切面编程, 最直接的体现就是 Spring 事物管理。至于 Spring 事物的相关资料, 就不说了, 参考: Spring 注解式事物管理

7、谈谈你对 NIO 的理解

答: IO 是面向流, NIO 是面向缓冲, 这里不细讲了, 具体参照: Java NIO 和 IO 的区别

8、ArrayList 和 LinkedList、Vector 的区别?

答: 总得来说可以理解为: .

1.ArrayList 是实现了基于动态数组的数据结构, LinkedList 基于链表的数据结构。

2.对于随机访问 get 和 set, ArrayList 觉得优于 LinkedList, 因为 LinkedList 要移动指针。

3.对于新增和删除操作 add 和 remove, LinedList 比较占优势, 因为 ArrayList 要移动数据

Vector 和 ArrayList 类似, 但属于强同步类, 即线程安全的, 具体比较参照: 比较 ArrayList、LinkedList、Vector

9、随便说说几个单例模式, 并选择一种线程安全的

答: 单例的类别: 懒汉、饿汉、枚举、静态内部类、双重校验锁 等等, 选择线程安全我选最后一种双重校验锁。具体实现方式参照: Java: 单例模式的七种写法

10、谈谈红黑树

答: 算法和数据结构一直是我薄弱之处, 这方面说自己补吧, 成效不大, 这里我就推荐一个: 红黑树

11、举例说说几个排序, 并说明其排序原理

答: 这里我就不细说了, 大家自己看看 Java 实现几种常见的排序算法

12、MySQL 索引的原理

答: 索引的作用大家都知道, 就是加快查询速度, 但是原理, 我说不上来, 这里直接看吧: MySQL 索引工作原理

13、序列化的原理和作用

答: Serialization (序列化) 是一种将对象以一连串的字节的描述的过程; 反序列化 deserialization 是种将这些字节重建成一个对象的过程, 主要用于 HTTP 或者 Webservice 接口传输过程中对象参数的播, 具体可参看: Java 序列化机制和原理

九、并发及项目调优

1、说说线程安全的几种实现方式?

答: 什么是线程安全? 我的理解是这样的, 一个对象被多个线程同时访问, 还能保持其内部属性的顺

性及同步性，则认定为线程安全。实现线程安全的三种方式：被 volatile、synchronized 等关键字修饰，或者使用 java.util.concurrent 下面的类库。至于前两者的关系，参考：synchronized 和 volatile 的用法区别

2、方法内部，如何实现更好的异步？

答：我们知道异步其实就是让另一个线程去跑，那么如何创建线程？第一种直接 new Thread，第二种 new 一个实现 Runnable 接口的实现类。第三种，通过线程池来管理创建等，这里说到更好的异步，那就是说我们在方法内部避免频繁的 new 线程，就可以考虑线程池了。那么线程池如何创建？这里可以 new 一个线程池，但是需要考虑单例，或者在程序初始启动时，就创建一个线程池，让他着，然后在具体方法的时候，通过线程池来创建线程，实现异步

3、项目中为何要用缓存？如何理解 nginx + Tomcat + Redis 集群缓存？

答 1：最直接的表现就是减轻数据库的压力。避免因为数据读取频繁或过大而影响数据库性能，降低序宕机的可能性

答 2：nginx 常用做静态内容服务和代理服务器，直面外来请求转发给后面的应用服务。nginx 本身能做缓存，比如静态页面的缓存什么的。而 Tomcat 是应用服务器，处理 Java Web 程序功能等等你也可以这么理解，假设把用户的请求当做是一条河流，那么 nginx 就相当于一个水利工程，Tomcat 相当于一条条分流的支流，而 Redis 相当于支流旁边的一个个水库。当你洪水来了，nginx 根据你条支流的承受力度分发不同的水流量，在确保程序正常运行的情况下，分发给每条支流(tomcat) 不的水流量。而 Redis 相当于一个个支流的水库，存储水源，降低压力，让后面的水量平稳。

4、日常项目中，如果你接手，你准备从哪些方面调优？

答：这个呢首先是了解哪些需要优化，需要优化肯定是项目性能遭遇瓶颈或者猜测即将遭遇了，我们会去考虑优化。那么怎么优化？

a、扩容，扩容的理解，就是扩充服务器并行处理的能力，简单来说就是加服务器，增加处理请求的力，例如增加 nginx、Tomcat 等应用服务器的个数，或者物理服务器的个数，还有加大服务器带宽等，这里考虑的是硬件方面

b、调优，调优，包括系统调优和代码调优。系统调优就是说加快处理速度，比如我们所提到的 CD、ehcache、Redis 等缓存技术，消息队列等等，加快服务间的响应速度，增加系统吞吐量，避免并，至于代码调优，这些就需要多积累了，比如重构、工厂等，数据库调优的话这个我不是很懂，只知道索引和存储过程，具体参考：MySQL 数据库调优 21 个最佳实践，其他数据库调优方面就各位自己找吧

5、谈谈你对分布式的理解

答：个人理解：分布式就是把一个系统/业务 拆分成多个子系统/子业务 去协同处理，这个过程就叫布式，具体的演变方式参考：Java 分布式应用技术架构介绍

6、Redis 实现消息队列

答：Redis 实现消息队列 、参考 2

7、另总结多线程相关面试题 50 道

8、分享一个调优工具和方案：如何利用 JConsole 观察分析 Java 程序的运行，进行排错调优

十、手写代码题（包含 SQL 题）

1、假设商户表 A (id, city)，交易流水表 B (aid, amount, time) 这里的 time 代表交易时间，请用 SQL 写出查询每个城市每个月的销售业绩（答案可在评论里回复）

2、假设有一个数组 A，int[] A = { 1, 3, -1, 0, 2, 1, -4, 2, 0, 1 ... N}; 原来是需要查出大于 0 数组，但是由于传参错误或者其他原因，导致查出 0 和负数了，现在要求在不使用新数组和新集合的情况下（即只使用这个 A 数组，因数组数据比较大，且只能用一次循环）实现正数放到数组的前面，于等于 0 的数放到数组的末尾（答案可在评论里回复）

十一、设计方案相关

面试还会问到一些关于设计方案相关的问题，比如

1、你的接口服务数据被人截包了，你如何防止数据恶意提交？

答：我们可以在接口传输参数里面设置一个业务编号，这个编号用来区分是否重复提交。这样即使被抓包了，对方也无法区分每个字段你的含义，这时，这个业务编号的作用就来了

2、假设服务器经常宕机，你从哪些方面去排查问题？

答：这个就留个各位看官补充了，可评论回复