

- 美团篇

5.1.0 java 虚拟机内存模型

5.1.1 内存溢出一般发生在哪个区？永久代会不会导致内存溢出？

5.1.2 动态加载类的框架了解哪些？

5.1.3 动态代理一般有哪几种实现方式？动态代理的应用场景有哪些？

5.1.4 栈会不会溢出？栈溢出一般抛什么异常？jvm 在哪里设置栈的大小？设置的参数是什么？

5.1.5 用过哪些命令查看 jvm 的状态、堆栈信息？

5.1.6 jvm 的垃圾回收机制？

5.1.7 java 类加载机制？如何实现自定义类加载器？`findClass` 与 `loadClass` 的区别？

5.1.8 `String`、`StringBuffer`、`StringBuilder` 的区别？对应的使用场景？

5.1.9 如何实现不可变的类？

5.2.0 浅复制和深复制？怎样实现深复制？

5.2.1 `HashMap`、`HashTable`、`ConcurrentHashMap` 的区别？

5.2.2 CAS 是一种什么样的同步机制？

5.2.3 NIO 的原理，包括哪几个组件？

5.2.4 简单介绍一下 java 的反射机制？反射在哪些地方有应用场景？

5.2.5 spring 加载 bean 的流程？

5.2.6 java 线程池？线程池构造函数的几个参数含义？`keepAliveTime` 解释一下？

5.2.7 一个接口，要去调用另外 5 个接口，每一个接口都会返回数据给这个调用接口，调用接口要对数据进行合并并返回给上层。这样一种场景可能用到并发包下的哪些类？你会怎么去实现这样的业务场景？

5.2.8 `CountDownLatch` 和 `CyclicBarrier` 的区别？

5.2.9 线程加锁有哪些方式？`synchronized` 和 `lock` 的区别？

5.3.0 `volatile` 关键字的作用？为什么使用 `AtomicLong` 而不使用 `Long`?`AtomicLong` 的底层是怎么实现的？

5.3.1 mysql 的存储引擎有哪几种？

5.3.2 sql 优化有哪些着手点？组合索引的最左前缀原则的含义？

5.3.3 springmvc 处理请求的流程？

5.3.4 spring 的事务怎么使用？事务回滚？自定义异常？

5.3.5 脏读？幻读？

5.3.6 tcp 四次挥手的过程？`TIME_WAIT` 为什么至少设置两倍的 MSL 时间？

5.3.7 `get` 和 `post` 请求的区别？

5.3.8 cookie 和 session 的请求？

5.3.9 了解哪些开源的中间件？缓存？消息？分布式框架？

5.4.0 用到过哪些设计模式？单例模式的实现？

5.4.1 数据库的事务实现原理、操作过程、如何做到事物之间的独立性等问题

5.4.2 数据库的脏读，幻读，不可重复读出现的原因原理，解决办法

5.4.3 数据库的隔离级别、MVCC

5.4.4 乐观锁、悲观锁、互斥锁、读写锁的原理实现与区别

5.4.5 线程的生命周期

5.4.6 一致性 hash 算法原理与应用

5.4.7 CAP 原则

5.4.8 CAS 操作

5.4.9 分布式 raft 算法

## 5.1.0 **java** 虚拟机内存模型

5.1.1 内存溢出一般发生在哪个区？永久代会不会导致内存溢出？

5.1.2 动态加载类的框架了解哪些？

5.1.3 动态代理一般有哪几种实现方式？动态代理的应用场景有哪些？

5.1.4 栈会不会溢出？栈溢出一般抛什么异常？**jvm** 在哪里设置栈的大小？设置的参数是什么？

5.1.5 用过哪些命令查看 **jvm** 的状态、堆栈信息？

5.1.6 **jvm** 的垃圾回收机制？

5.1.7 **java** 类加载机制？如何实现自定义类加载器？**findClass** 与 **loadClass** 的区别？

5.1.8 **String**、**StringBuffer**、**StringBuilder** 的区别？对应的使用场景？

5.1.9 如何实现不可变的类？

5.2.0 浅复制和深复制？怎样实现深复制？

5.2.1 **HashMap**、**HashTable**、**ConcurrentHashMap** 的区别？

5.2.2 **CAS** 是一种什么样的同步机制？

5.2.3 **NIO** 的原理，包括哪几个组件？

5.2.4 简单介绍一下 **java** 的反射机制？反射在哪些地方有应用场景？

5.2.5 **spring** 加载 **bean** 的流程？

5.2.6 **java** 线程池？线程池构造函数的几个参数含义？**keepAliveTime** 解释一下？

5.2.7 一个接口，要去调用另外 5 个接口，每一个接口都会返回数据给这个调用接口，调用接口要对数据进行合并并返回给上层。这样一种场景可能用到并发包下的哪些类？你会怎么去实现这样的业务场景？

5.2.8 **CountDownLatch** 和 **CyclicBarrier** 的区别？

5.2.9 线程加锁有哪些方式？**synchronized** 和 **lock** 的区别？

5.3.0 **volatile** 关键字的作用？为什么使用 **AtomicLong** 而不使用 **Long?AtomicLong** 的底层是怎么实现的？

5.3.1 **mysql** 的存储引擎有哪几种？

5.3.2 **sql** 优化有哪些着手点？组合索引的最左前缀原则的含义？

5.3.3 **springmvc** 处理请求的流程？

5.3.4 **spring** 的事务怎么使用？事务回滚？自定义异常？

5.3.5 脏读？幻读？

5.3.6 **tcp** 四次挥手的过程？**TIME\_WAIT** 为什么至少设置两倍的 **MSL** 时间？

5.3.7 **get** 和 **post** 请求的区别？

5.3.8 **cookie** 和 **session** 的请求？

5.3.9 了解哪些开源的中间件？缓存？消息？分布式框架？

5.4.0 用到过哪些设计模式？单例模式的实现？

**5.4.1** 数据库的事务实现原理、操作过程、如何做到事物之间的独立性等问题

**5.4.2** 数据库的脏读，幻读，不可重复读出现的原因原理，解决办法

**5.4.3** 数据库的隔离级别、MVCC

**5.4.4** 乐观锁、悲观锁、互斥锁、读写锁的原理实现与区别

**5.4.5** 线程的生命周期

**5.4.6** 一致性 hash 算法原理与应用

**5.4.7** CAP 原则

**5.4.8** CAS 操作

**5.4.9** 分布式 raft 算法