## ConcurrentHashmap实现线程安全的原理

jdk1.7中是采用分段锁（segment）的方式实现，每个分段锁类似hashmap的实现，对写加锁，默认为16个。

jdk1.8中，它摒弃了Segment（锁段）的概念，而是启用了一种全新的方式实现,利用CAS算法。它沿用了与它同时期的HashMap版本的思想，底层依然由“数组”+链表+红黑树的方式思想([JDK7与JDK8中HashMap的实现](http://my.oschina.net/hosee/blog/618953))。

## hashtable实现线程安全的原理

加了读写锁实现线程安全

## 你是如何进行高并发编程的

1.比方说计数时，为了保证原子性，都会采用jdk封装的原子对象AtomicInteger

2.数据存储上，使用concurrentHashmap替代hashmap

3.需要异步执行时，多采用全局的线程池对象，做到线程重用，减少创建线程的开销

## 你是如何做jvm调优的

 确保对象都是“朝生夕死”的   
一个对象使用完后应尽快让他失效，然后尽快在新生代中被Minor GC回收掉，尽量避免对象在新生代中停留太长时间。

 提高大对象直接进入老年代的门槛   
通过设置参数-XX:PretrnureSizeThreshold来提高大对象的门槛，尽量让对象都先进入新生代，然后尽快被Minor GC回收掉，而不要直接进入老年代。

 编码上尽量避免使用大对象

## 你的自研框架主要是做了哪些优化

http请求处理、中间件集群统一api，日志追踪，异常返回统一处理，mybatis加载机制适配、指标统计、

## ArrayList扩容原理

超过容量限制时，每次以原来的1.5倍扩容

## 什么情况下会产生死锁

资源竞争，需要锁的对象被其他线程持有，同事其他线程也处于这个状态，一直无法获取到锁，导致一直挂起。

## 操作系统的四个特性

并发：同一段时间内多个程序执行(注意区别并行和并发，前者是同一时刻的多个事件，后者是同一时间段内的多个事件)  
共享：系统中的资源可以被内存中多个并发执行的进线程共同使用  
虚拟：通过时分复用（如分时系统）以及空分复用（如虚拟内存）技术实现把一个物理实体虚拟为多个  
异步：系统中的进程是以走走停停的方式执行的，且以一种不可预知的速度推进

## 锁产生的四个必要条件

* 互斥条件：资源是独占的且排他使用，进程互斥使用资源，即任意时刻一个资源只能给一个进程使用，其他进程若申请一个资源，而该资源被另一进程占有时，则申请者等待直到资源被占有者释放。
* 不可剥夺条件：进程所获得的资源在未使用完毕之前，不被其他进程强行剥夺，而只能由获得该资源的进程资源释放。
* 请求和保持条件：进程每次申请它所需要的一部分资源，在申请新的资源的同时，继续占用已分配到的资源。
* 循环等待条件：在发生死锁时必然存在一个进程等待队列{P1,P2,…,Pn},其中P1等待P2占有的资源，P2等待P3占有的资源，…，Pn等待P1占有的资源，形成一个进程等待环路，环路中每一个进程所占有的资源同时被另一个申请，也就是前一个进程占有后一个进程所深情地资源。   
  以上给出了导致死锁的四个必要条件，只要系统发生死锁则以上四个条件至少有一个成立。事实上**循环等待**的成立蕴含了前三个条件的成立，似乎没有必要列出然而考虑这些条件对死锁的预防是有利的，因为可以通过破坏四个条件中的任何一个来预防死锁的发生。

## 索引是什么？有什么作用以及优缺点？

索引是对数据库表中一或多个列的值进行排序的结构，是帮助MySQL高效获取数据的数据结构

你也可以这样理解：索引就是加快检索表中数据的方法。数据库的索引类似于书籍的索引。在书籍中，索引允许用户不必翻阅完整个书就能迅速地找到所需要的信息。在数据库中，索引也允许数据库程序迅速地找到表中的数据，而不必扫描整个数据库。

**MySQL**数据库几个基本的索引类型：普通索引、唯一索引、主键索引、全文索引

* 索引加快数据库的检索速度
* 索引降低了插入、删除、修改等维护任务的速度
* 唯一索引可以确保每一行数据的唯一性
* 通过使用索引，可以在查询的过程中使用优化隐藏器，提高系统的性能
* 索引需要占物理和数据空间

## 使用索引查询一定能提高查询的性能吗？为什么

通常,通过索引查询数据比全表扫描要快.但是我们也必须注意到它的代价.

索引需要空间来存储,也需要定期维护, 每当有记录在表中增减或索引列被修改时,索引本身也会被修改. 这意味着每条记录的INSERT,DELETE,UPDATE将为此多付出4,5 次的磁盘I/O. 因为索引需要额外的存储空间和处理,那些不必要的索引反而会使查询反应时间变慢.使用索引查询不一定能提高查询性能,索引范围查询(INDEX RANGE SCAN)适用于两种情况:

## 说说数据库的三大范式（个人表述）

**第一范式：属性不可分割**

**第二范式：要有主键**

**第三范式，消除冗余**

## 什么是事务？

事务（Transaction）是并发控制的基本单位。所谓的事务，它是一个操作序列，这些操作要么都执行，要么都不执行，它是一个不可分割的工作单位。事务是数据库维护数据一致性的单位，在每个事务结束时，都能保持数据一致性。

## Springmvc实现原理

2、  DispatcherServlet收到请求调用HandlerMapping处理器映射器。

3、  处理器映射器找到具体的处理器(可以根据xml配置、注解进行查找)，生成处理器对象及处理器拦截器(如果有则生成)一并返回给DispatcherServlet。

4、  DispatcherServlet调用HandlerAdapter处理器适配器。

5、  HandlerAdapter经过适配调用具体的处理器(Controller，也叫后端控制器)。

6、  Controller执行完成返回ModelAndView。

7、  HandlerAdapter将controller执行结果ModelAndView返回给DispatcherServlet。

8、  DispatcherServlet将ModelAndView传给ViewReslover视图解析器。

9、  ViewReslover解析后返回具体View。

10、DispatcherServlet根据View进行渲染视图（即将模型数据填充至视图中）。

11、 DispatcherServlet响应用户。

## nio和bio的区别

* BIO：同步阻塞式IO，服务器实现模式为一个连接一个线程，即客户端有连接请求时服务器端就需要启动一个线程进行处理，如果这个连接不做任何事情会造成不必要的线程开销，当然可以通过线程池机制改善。
* NIO：同步非阻塞式IO，服务器实现模式为一个请求一个线程，即客户端发送的连接请求都会注册到多路复用器上，多路复用器轮询到连接有I/O请求时才启动一个线程进行处理。

你在开发中遇到的比较大的困难？