### 多线程在项目中的使用场景？

#### 答案

这个太多了。用户请求、数据库连接、轮询任务，端口监听，异步操作等等等等。只要操作涉及到性能和计算，全部要用到多线程

#### 分析

很常见的多线程问题，对于多线程求职者一定要重视，顺着这个问题，面试官可能会问到多线程的实现原理，为什么能提高效率，对于这方面要做好准备

### 2.说一下gc回收机制的原理？

用户Java程序运行过程中，Java虚拟机提供了另外一个系统级的线程，专门负责回收不再被使用的对象占用的内存，这一过程称为垃圾回收。垃圾回收需要对堆内存中的对象进行标记，并对堆内存进行整理。这一过程的某些阶段需要暂时终止用户Java线程，等回收工作完成后再恢复执行。因此，频繁地触发虚拟机垃圾回收操作的行为会影响程序的运行效率。那么什么情况下会频繁地出发垃圾回收操作呢？- 比如：堆内存设置过小- 再比如：程序频繁地分配大型局部对象数组

#### 分析：

在有些公司的面试中，可能会问到一些虚拟机的底层原理，而gc回收机制就是常问的一种，当问及这类问题的时候，面试官实际上是想考察求职者对底层原理的了解程度，从而推断出作为一名程序员的自主学习能力，在平时开发中，在敲代码之外，是否有时间去研究代码的源码，和JVM的原理，如果想了解JVM原理和机制方面的知识，推荐大家看《深入了解Java虚拟机》。

#### 拓展：

涉及到底层原理，还有一种常考的方式就是，虚拟机的内存划分，堆、栈、方法区等，所以面试者要对这些底层概念分清楚，明确他们存储的是什么，以及堆栈的特点。

### 3.说说JDK1.8的新特性？

#### 答案：

**Lambda 表达式** − Lambda允许把函数作为一个方法的参数（函数作为参数传递进方法中。

**方法引用** − 方法引用提供了非常有用的语法，可以直接引用已有Java类或对象（实例）的方法或构造器。与lambda联合使用，方法引用可以使语言的构造更紧凑简洁，减少冗余代码。

**默认方法** − 默认方法就是一个在接口里面有了一个实现的方法。

**新工具** − 新的编译工具，如：Nashorn引擎 jjs、 类依赖分析器jdeps。

**Stream API** −新添加的Stream API（java.util.stream） 把真正的函数式编程风格引入到Java中。

**Date Time API** − 加强对日期与时间的处理。

**Optional 类** − Optional 类已经成为 Java 8 类库的一部分，用来解决空指针异常。

**Nashorn, JavaScript 引擎** − Java 8提供了一个新的Nashorn javascript引擎，它允许我们在JVM上运行特定的javascript应用。

#### 分析：

我们在踏入这个行业之后，要关注这个行业的新动态，面试官之所以会问这个问题，就是在考察求职者对于新鲜事物或者新技术的接受能力，有没有对java的新动向有了解。

### 4.说一说Tomcat优化方式？

#### 答案

Tomcat的优化我准备从三方面来说:

第一部分: 内存优化

Tomcat的默认内存配置比较低,不用说大项目,就算是小项目,并发量达到一定程度也就可能会抛出OutOfMemoryError异常,为了解决这个问题,我们要修改JVM的一些配置,在tomcat的bin目录下的catalina配置文件中,配置Xms和Xmx,也就是

Java虚拟机初始化时堆的最小内存和最大内存,这俩值通常会配置成一样,这样GC不必再为扩展内存空间而消耗性能.除了这两个,还可以配置XX:PermSize和XX:MaxPermSize,它们是Java虚拟机永久代大小和最大值,除了这几个参数

还可以再根据具体需要配置其他参数。

第二部分: 配置优化

配置优化,主要有三方面:

1. Connector 优化

Connector是连接器，它负责接收客户的请求，以及向客户端回送响应的消息。

默认情况下Tomcat只支持200线程访问，超过这个数量的连接将被等待甚至超时放弃，所以我们需要提高这方面的处理能力.

修改这部分配置需要修改conf下的server.xml,找到Connector 标签项,修改protocol,默认的协议类型是BIO,也就是阻塞式I/O操作,

简单项目及应用可以采用BIO.

第二种协议类型是NIO,它就是一种基于缓冲区是、并能提供非阻塞I/O操作的java API,它有更好的并发运行性能. NIO更适合后台需要耗时完成请求的操作

第三种协议类型是APR,它主要可以提高Tomcat对静态文件的处理性能.

选择哪个协议也是根据实际项目进行配置.

除了这个协议类型,还有一个非常重要的参数要改,就是maxThreads,就是当前连接器能够处理同时请求的最大数目.这个数目也并非越大越好,它也受操作系统等硬件制约,所以这个值要根据压力测试后实际数据进行配置.

2. 线程池

使用线程池的好处在于减少了创建销毁线程的相关消耗，而且可以提高线程的使用效率。使用线程池就在Service标签中配置Executor就可以了

3. Listener

还有一个影响tomcat性能的因素是内存泄漏,我们在Server标签中配置一个JreMemoryLeakPreventionListener就可以用来预防JRE内存泄漏

第三部分: 组件优化

可以选用Tomcat Native组件，它可以让 Tomcat使用 Apache 的 APR包来处理包括文件和网络IO操作，从而提升性能及兼容性

#### 分析

面试官问到这个问题其实是考察的比较深，初入职场的Java程序员是接触不到的。可以选择答案里的几种方案进行陈述。

### 5.Sql语句优化有哪些？数据库的优化怎样优化？

#### 答案

1. 对查询进行优化，应尽量避免全表扫描，首先应考虑在 where 及 order by 涉及的列上建立索引。

2. 应尽量避免在 where 子句中对字段进行 null 值判断，否则将导致引擎放弃使用索引而进行全表扫描

3. 应尽量避免在 where 子句中使用!=或<>操作符，否则引擎将放弃使用索引而进行全表扫描。

### 6.如何解决数据高并发？流程是怎么样的？如何保证高效率处理高并发？

#### 答案：

1、尽量将请求的页面静态化 静态化的页面为.html(.htm等)不需要web服务器重新加载项解析，只需要生成一次，以后每次都直接下载到客户端，效率高很多。javaWeb静态化的技术有freemark和Velocity等。

2、fastDFS图片服务器：

将网站系统的web服务器、数据库服务器、图片和文件服务器分开，通过将服务器专业化分工，以提高网站访问速度。因为图片和文件在下载的时候无论是IIS、Apache等服务器都会有很大压力。

3、数据缓存服务器 redis

可以设置专门的数据缓存服务器，将大量数据放到缓存数据区，在访问量少得时候存入数据，减少连接直接操作数据库的开销。

4、数据库集群、库表散列（数据库的各种优化、数据库的拆分）

5、镜像

镜像是大型网站常采用的提高性能和数据安全性的方式，镜像的技术可以解决不同网络接入商和地域带来的用户访问速度差异，比如ChinaNet和EduNet之间的差异就促使了很多网站在教育网内搭建镜像站点，数据进行定时更新或者实时更新。在镜像的细节技术方面，这里不阐述太深，有很多专业的现成的解决架构和产品可选。也有廉价的通过软件实现的思路，比如Linux上的rsync等工具。

6、负载均衡（nginx）

负载均衡将是大型网站解决高负荷访问和大量并发请求采用的高端解决办法。

7、最新：CDN加速技术（这个比镜像更好用）

CDN的全称是内容分发网络。其目的是通过在现有的Internet中增加一层新的网络架构，将网站的内容发布到最接近用户的网络“边缘”，使用户可以就近取得所需的内容，提高用户访问网站的响应速度。

### 7.首页展示的轮播图，在页面中是如何存放的？在数据库中是如何存放的？

#### 答案

轮播图在前端页面中是存在数组或json中

在数据库中是存的轮播图的指向地址

### 8. 商品上下架如何实现前后台消息同步？

#### 答案

在分布式缓存中设置轮询位。例如在分布式缓存中，某个节点Prodct1: Normal表示商品1正常，后台订阅该位置，当值发生变动，回调业务操作即可。

当然也有复杂的，用MQ、AKKA等机制来通知等等

### 9. 消息中间件activeMQ中丢了消息你们是怎么处理的？

#### 答案

查日志，查死信队列，查持久化文件等等把丢失的数据找回来。消费方订阅方式不要使用“非持久化订阅”的方式，如果不是太重要，发出预警，通知生产者重新发送。

另外，有条件的话，需要搭建日志分析服务；将生产上的日志与MQ日志等进行统计分析，例如时间比对，域比对等等，综合计算消息完成率；以及失败消息的分析。

#### 分析

很实际的项目开发问题，面试中不仅会问到技术点的原理，怎样应用，更多的会问到异常处理

### 10.分布式系统中A服务有两台服务器，那么我们在调用的时候如何知道是调用的那台服务器？（dubbo可以实现，那么不用dubbo能解决吗？）

#### 答案

调用远程服务的时候，假设是RPC服务，那么通常的算法对于调用方来讲是未知的。如果已知的情况只有在知道分发策略的情况下，也就是说，你知道程序会怎么调的情况下。

从技术角度来讲，底层无非就是协议栈远程访问，那么IP肯定是透明的，只要开发代码能够接触到NET层。

另外还有一种结构，服务注册中心实时监控服务器某个服务 调用次数/时间 这样的一个负载值，会根据不同的负载值给客户端最合理的建议（有时候仅仅是建议），而客户端再根据某些算法或者约定再确定调用哪台机器上的服务。

总之一句话，RPC服务调用之前会确定将会调用服务的宿主机的IP、端口，协议栈等等信息，如果能够接触到这部分核心代码，便能知道具体的网络信息。

#### 分析

多线程的锁对象在开发中的具体应用，又是这种问题，面试官会把开发中可鞥出现问题的情况怎么预处理，和已经出现了情况怎么处理向求职者询问，这种很实际的问题即考察面试者的技术功底又能考验面试者的思路和应变能力，越来越多的公司会采取这种题替代死板的知识点问答，所以公司对入职要求的升级，导致求职者对自身的各方面条件也要升级，不止是技术

### 11.库存减少是当生成订单减少的，那么假设A商品只有两件了，甲和乙同时点击了生成订单，那么谁能下单成功？谁不能成功？怎么实现的？

#### 答案

这个是简单的锁问题（秒杀问题）。

假设库存只有2件，AB同时下单，那么AB两个用户将竞争锁资源。例如，两件商品生成锁L1，L2；A用户所在线程获取到了L1，这时上下文切换到B所在线程，它发现L1已经被占了，至于有没有下单完成还不知道，这时它尝试获取L2，并进行后续操作。假设这时用户A不想买了，或者支付超时，那么L1将会被释放。

### 12.未登录时购物车信息，存在cookie中？还有哪些方式？

#### 答案

未登录状态下,购物车信息就是存在cookie中的,如果禁用cookie的话,servlet引入了一种补充会话的机制 , 当用户发出下一次请求时 , 如果请求信息中没有包含Cookie头字段 , Servlet引擎则认为客户端不支持cookie,它将依据请求url参数中的会话标识号来实施会话跟踪.

这种技术称为url重写，tomcat发送给客户端的会话标识号的Cookie名称为JSESSIONID。url重写就是将JSESSIONID关键字作为参数名和会话标识号作为参数附加到URL后面。如果浏览器不支持Cookie或者关闭Cookie，就必须对所有可能被客户端访问的请求路径进行 URL重写，如超链接，form表单的action属性和重定向的URL.

response.encodeURL() 使 URL包含 session ID，如果你需要使用重定向，可以使用 response.encodeRedirectURL ()来对 URL 进行编码。encodeURL () 及 encodeRedirectedURL () 方法首先判断 cookie是否被浏览器支持；如果支持，则参数 URL 被原样返回，session ID 将通过 cookies 来维持。

#### 分析

这个问题是比较高端的问题，但凡遇到这种难度的问题，要么功底深厚掌握的特别明白，也能说的很清楚，如果知识一知半解的，最好不要说，碰巧面试官对这个问题很熟悉，那这次面试很容易挂掉，会给面试官留下不实在的感受，所以任何时候对任何问题的回答都要随机应变，相时而动

### 13.Nginx实现Tomcat集群的负载均衡有几种方式，你们项目中使用的是那种方式？

#### 答案：

一，如果仅是对外提供一个页面访问，不用区分单一用户（不区分每个访问session，不涉及用户权限，用户资料等内容），仅仅配置nginx负载均衡策略即可。

nginx负载均衡策略主要分一下四种：

1) 轮询（默认）

每个请求按时间顺序逐一分配到不同的后端服务器，如果后端服务器宕机，能自动剔除。

2) ip\_hash 每个请求按访问ip的hash结果分配，这样每个访客固定访问一个后端服务器。

3) fair 按后端服务器的响应时间来分配请求，响应时间短的优先分配。

4) url\_hash 按访问url的hash结果来分配请求，使每个url定向到同一个后端服务器，后端服务器为缓存时比较有效。

二，如果涉及到用户session，做一些鉴权缓存、存放临时信息时，就必须做tomcat的session共享。

目前可参考到的session共享方式主要分为两种。

1）利用tomcat自带的组播机制，实现session复制。

对tomcat及应用的若干配置文件进行配置即可实现，网上有很多资料可参考。但这种方式些弊端，看过一些资料，不建议用session复制的方式。在实际使用过程中，也发现有存在session莫名失踪的现象。

2）利用第三方机制存储session。

比较常见的是tomcat集成memcached服务器来存储session。实际项目中，我们采用过利用redis实现session存储，redis高效的存取性能为高效的访问提供了保障，但是目前redis的集群功能似乎没有发布，如何解决redis的单点故障需要研究。

### 14.Redis如何实现秒杀业务？

#### 答案

1、把商品的数量放到 redis 中。

2、秒杀时使用 decr 命令对商品数量减一。如果不是负数说明抢到。

3、一旦返回数值变为 0 说明商品已售完

### 15.项目中是如何解决Session共享的？

#### 答案

（1）session复制

tomcat的session复制，可以实现session共享

优点：不需要额外开发，只需搭建tomcat集群即可

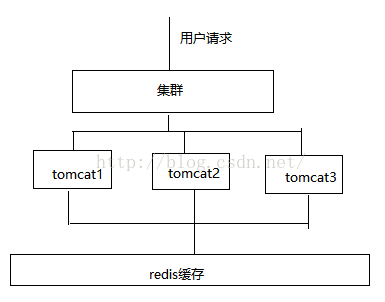
缺点：tomcat 是全局session复制，集群内每个tomcat的session完全同步（也就是任何时候都完全一样的) 在大规模应用的时候，用户过多，集群内tomcat数量过多，session的全局复制会导致集群性能下降， 因此，tomcat的数量不能太多，5个以下为好。

（2）session绑定

当用户A第一次访问系统时，tomcat1对其进行服务，那么，下次访问时仍然让tomcat1对其进行服务

（3）使用redis集中管理session

可以将用户的会话保存在redis中，每次从redis中查询用户信息，就可以很好的解决会话共享问题。如图：



四、实际应用

（1）用户登录问题

对于大型分布式系统，可以使用单点登录系统进行登录，其中用户的session保存在redis缓存系统中

（2）用户短信验证

当需要对用户短信进行校验前，调取第三方服务获取验证码，需要先将验证码保存在session中，然后与用户提交的验证码进行比对

### 16.solr全文检索，如果有敏感词汇，如何处置？

### 17.spring框架的aop和ioc在项目中的应用场景？

#### 答案

AOP 用来封装横切关注点，具体可以在下面的场景中使用:

Authentication 权限、Caching 缓存、Context passing 内容传递、Error handling 错误处理Lazy loading 懒加载、Debugging 调试、logging, tracing, profiling and monitoring 记录跟踪优化校准、Performance optimization 性能优化、Persistence 持久化、Resource pooling 资源池、Synchronization 同步、Transactions 事务

原理：AOP 是面向切面编程，是通过代理的方式为程序添加统一功能，集中解决一些公共问题。

好处：1.各个步骤之间的良好隔离性2.源代码无关性

IOC: 当某个角色需要另外一个角色协助的时候，在传统的程序设计过程中，通常由调用者来创建被调用者的实例。但在 spring 中创建被调用者的工作不再由调用者来完成，因此称为控制反转。创建被调用者的工作由 spring 来完成，然后注入调用者 。人IOC 本质上是一个容器，已 MAP 对 IOC 简单举例，服务器加载配置文件，由 xml 文档解析工具读取 bean 的 ID，获取 class，使用反射创建对象，以 K-V 的形式存入 MAP，K 是 ID，V 是反射创建的对象。获取对象可以调用 context.getBean（K）的方式。

### 18.ActiveMQ：说说activeMQ调用流程结合模块说一下

#### 答案：

发送消息的基本步骤：

（1）创建连接使用的工厂类 JMS ConnectionFactory。

（2）使用管理对象JMS ConnectionFactory建立连接Connection，并启动。

（3）使用连接Connection建立会话Session。

（4）使用会话Session和管理对象Destination创建消息生产者MessageSender。

（5）使用消息生产者MessageSender发送消息

消息接收者从JMS接收消息的步骤：

（1）创建连接使用的工厂类JMS ConnectionFactory。

（2）使用管理对象JMS ConnectionFactory建立连接Connection，并启动。

（3）使用连接Connection建立会话Session。

（4）使用会话Session和管理对象Destination创建消息接收者MessageReceiver。

（5）使用消息接收者MessageReceiver接收消息，需要用setMessageListener将MessageListener接口绑定到MessageReceiver，消息接收者必须实现了MessageListener接口，需要定义onMessage事件方法。

### 19.Redis挂掉后怎么办?介绍下redis是怎么实现高可用的?

#### 答案：

主要取决于，你是把redis作为缓存还是nosql，如果是缓存那丢了也无所谓，从别的地方恢复重建就行了，如果是nosql的话，redis是有snapshot和aof的机制来保证数据持久化的。另一个方面是否搭建了redis集群，若是主机宕机，可以使用备机。

### 20.如果项目已经上线了你，但是出现了问题。主要是怎么解决，或者你们怎样找出问题所在的（日志方向），这块你接触过吗？

#### 答案：

通过引用log4j日志包来查看，在每个类中引用相应的日志来记录，在服务器中留出专门的空间来存储日志，后缀为.log的文件，出现的错误会存储到日志文件中，当我们需要查看的时候通过编译软件来查看。实际开发中，这一块一般都是运维人员负责的。

#### 分析：

在我们的实际开发中，项目一般都是运行在服务器上的，经常会有一些错误出现，而这些错误通常会保存在日志文件中，面试官问这道题的意义在于考察你是否有实际的开发经验和解决办法

#### 拓展：

接下来可能会问到LINUX的常用命令，如何去查看日志的命令