**1.Netty 是什么？**

Netty 是一款基于 NIO（Nonblocking I/O，非阻塞IO）开发的网络通信框架，对比于 BIO（Blocking I/O，阻塞IO），他的并发性能得到了很大提高。难能可贵的是，在保证快速和易用性的同时，并没有丧失可维护性和性能等优势。

**2.Netty 的特点是什么？**

高并发：Netty 是一款基于 NIO（Nonblocking IO，非阻塞IO）开发的网络通信框架，对比于 BIO（Blocking I/O，阻塞IO），他的并发性能得到了很大提高。

传输快：Netty 的传输依赖于零拷贝特性，尽量减少不必要的内存拷贝，实现了更高效率的传输。

封装好：Netty 封装了 NIO 操作的很多细节，提供了易于使用调用接口。

**3.什么是 Netty 的零拷贝？**

Netty 的零拷贝主要包含三个方面：

Netty 的接收和发送 ByteBuffer 采用 DIRECT BUFFERS，使用堆外直接内存进行 Socket 读写，不需要进行字节缓冲区的二次拷贝。如果使用传统的堆内存（HEAP BUFFERS）进行 Socket 读写，JVM 会将堆内存 Buffer 拷贝一份到直接内存中，然后才写入 Socket 中。相比于堆外直接内存，消息在发送过程中多了一次缓冲区的内存拷贝。

Netty 提供了组合 Buffer 对象，可以聚合多个 ByteBuffer 对象，用户可以像操作一个 Buffer 那样方便的对组合 Buffer 进行操作，避免了传统通过内存拷贝的方式将几个小 Buffer 合并成一个大的 Buffer。

Netty 的文件传输采用了 transferTo 方法，它可以直接将文件缓冲区的数据发送到目标 Channel，避免了传统通过循环 write 方式导致的内存拷贝问题。

**4.Netty 的优势有哪些？**

使用简单：封装了 NIO 的很多细节，使用更简单。

功能强大：预置了多种编解码功能，支持多种主流协议。

定制能力强：可以通过 ChannelHandler 对通信框架进行灵活地扩展。

性能高：通过与其他业界主流的 NIO 框架对比，Netty 的综合性能最优。

稳定：Netty 修复了已经发现的所有 NIO 的 bug，让开发人员可以专注于业务本身。

社区活跃：Netty 是活跃的开源项目，版本迭代周期短，bug 修复速度快。

**5.Netty 的应用场景有哪些？**

典型的应用有：阿里分布式服务框架 Dubbo，默认使用 Netty 作为基础通信组件，还有 RocketMQ 也是使用 Netty 作为通讯的基础。

6.Netty 高性能表现在哪些方面？

IO 线程模型：同步非阻塞，用最少的资源做更多的事。

内存零拷贝：尽量减少不必要的内存拷贝，实现了更高效率的传输。

内存池设计：申请的内存可以重用，主要指直接内存。内部实现是用一颗二叉查找树管理内存分配情况。

串形化处理读写：避免使用锁带来的性能开销。

高性能序列化协议：支持 protobuf 等高性能序列化协议。