1. jdk动态代理与cglib代理的区别？

Jdk动态代理只能对实现了接口的类进行代理，而不能针对类；cglib代理针对类实现代理，对指定的类生成子类；

Jdk动态代理是利用反射机制生成一个实现代理接口的匿名类，而cglib代理是利用asm开源包，对代理对象类的class文件加载进来，通过修改其字节码生成子类来处理

1. jvm的调优命令有哪些？

Jps、jstas、jmap、jinfo

1. java会存在内存泄漏嘛？

会，长生命周期的对象持有短生命周期的对象的引用就很可能发生内存泄漏，尽管短生命周期对对象已经不在需要，但是因为长生命周期对象持有它的引用而导致不能回收，这就是java中内存泄漏的场景

1. 什么是双亲委派模式？如何打破？

双亲委派模式：如果一个类加载器需要加载一个类，那么首先它会把这个类请求委派给父类加载器去完成，每一层都是如此，一直递归到顶层，当父类加载器无法完成这个请求时，子类才会去尝试加载。

如何打破：写一个类加载器，继承classloader类，再重写loadclass和findclass方法

1. 为什么jdk8将永久代替换成元空间？

永久代在方法区中容易出现内存溢出，永久代会为GC带来不必要的复杂度，并且回收效率偏低。

1. string类型是如何实现不可变的？

首先，string是用final修饰的char数组，用final修饰就意味着必须在构造时为其赋值，且赋值后引用不能再改变，其次是string并没有提供接口来改变其value的值，value的值就不能获取。然而并不是真正的不能获取，可通过反射获取其内部属性。

1. hashCode()与equals()？

HashCode（）作用是获取哈希码，从而确实数据在哈希表中的索引位置，这只在哈希容器内才有意义。equals判断两个对象是否相等。Equals想等的话，则hashcode也相等。

1. 重写equals时为什么也要重写hashcode？

默认从object继承过来的hashcode是基于对象的id实现的；重写euqals是为了两个对象能够按我们自己的想法来判断是否相等，如果不重写hashcode方法，则会出现具有相同含义的不同对象的情况。