**1、String s = new String("xyz");创建了几个字符串对象？**

答：两个对象，一个是静态区的"xyz"，一个是用new创建在堆上的对象。

**2、接口是否可继承（extends）接口？抽象类是否可实现（implements）接口？抽象类是否可继承具体类（concrete class）？**

答：接口可以继承接口，而且支持多重继承。抽象类可以实现(implements)接口，抽象类可继承具体类也可以继承抽象类。

**3、一个".java"源文件中是否可以包含多个类（不是内部类）？有什么限制？**

答：可以，但一个源文件中最多只能有一个公开类（public class）而且文件名必须和公开类的类名完全保持一致。

**4、Anonymous Inner Class(匿名内部类)是否可以继承其它类？是否可以实现接口？**

答：可以继承其他类或实现其他接口，在Swing编程和Android开发中常用此方式来实现事件监听和回调。

**5、内部类可以引用它的包含类（外部类）的成员吗？有没有什么限制？**

答：一个内部类对象可以访问创建它的外部类对象的成员，包括私有成员。

**6、Java 中的final关键字有哪些用法？**

答：

　　(1)修饰类：表示该类不能被继承；

　　(2)修饰方法：表示方法不能被重写；

　　(3)修饰变量：表示变量只能一次赋值以后值不能被修改（常量）。

**7、指出下面程序的运行结果。**

class A {

static {

System.out.print("1");

}

public A() {

System.out.print("2");

}

}

class B extends A {

static {

System.out.print("a");

}

public B() {

System.out.print("b");

}

}

public class Hello {

public static void main(String[] args) {

A ab = new B();

ab = new B();

}

}

答：执行结果：1a2b2b。创建对象时构造器的调用顺序是：先初始化静态成员，然后调用父类构造器，再初始化非静态成员，最后调用自身构造器。

**提示**：如果不能给出此题的正确答案，说明之前第21题Java类加载机制还没有完全理解，赶紧再看看吧。

**8、数据类型之间的转换**

如何将字符串转换为基本数据类型？

答：调用基本数据类型对应的包装类中的方法parseXXX(String)或valueOf(String)即可返回相应基本数据类型。

如何将基本数据类型转换为字符串？

答：一种方法是将基本数据类型与空字符串（""）连接（+）即可获得其所对应的字符串；另一种方法是调用String 类中的valueOf()方法返回相应字符串。

**9、如何实现字符串的反转及替换？**

答：方法很多，可以自己写实现也可以使用String或StringBuffer/StringBuilder中的方法。有一道很常见的面试题是用递归实现字符串反转，代码如下所示：

public static String reverse(String originStr) {

if (originStr == null || originStr.length() <= 1)

return originStr;

return reverse(originStr.substring(1)) + originStr.charAt(0);

}

**10、怎样将GB2312编码的字符串转换为ISO-8859-1编码的字符串？**

答：代码如下所示：

String s1 = "你好";

String s2 = new String(s1.getBytes("GB2312"), "ISO-8859-1");