1.什么是Java虚拟机？为什么Java被称作是“平台无关的编程语言”？

Java虚拟机（JVM）是运行所有Java程序的抽象计算机，是Java语言的运行环境。Java虚拟机包括一套字节码指令集、一组寄存器、一个栈、一个垃圾回收堆和一个存储方法域。只要根据JVM规格描述将解释器移植到特定的计算机上，就能保证经过编译的任何Java代码能够在该系统上运行

平台无关有两种：源代码级和目标代码级。Java主要依靠JVM 在目标代码级实现平台无关性。因为JVM屏蔽了与具体平台相关的信息，使得Java编译时只需要生成在JVM上运行的目标代码，由虚拟机在具体平台上运行，从而实现一次编译，处处执行。。另外，Java采用的是基于IEEE标准的数据类型。通过JVM保证数据类型一致性，也确保了Java的平台无关性。

2.JDK和JRE的区别是什么？

JDK是Java开发工具包，是整个Java的核心，包括JRE（Java运行环境）、一堆Java工具和Java基础的类库。JRE是一个运行环境，其中包含JVM。具备开发功能的JDK中的JRE下同时有client性质的JVM和serve性质的JVM，仅仅作为运行环境的JRE下只有client性质的JVM。

3.Java支持的数据类型有哪些？什么是自动拆装箱？

八种基本类型：byte（字节类型）、short（短整型）、int（整数类型）、long（长整型）、float（单精度浮点类型）、double（双精度浮点型）、boolean（布尔变量）、Char（字符型）。

三种引用类型：class（类）、interface（接口）、array（数组）。

自动装箱是Java自动将原始类型转换成对应的对象，如将int的变量转换成Integer对象，反之将Integer对象转换成int类型值叫拆箱。原始类型byte、short、char、int、long、double和boolean对应的封装类为Byte、Short、Character、Integer、Long、Float、Double、Boolean。

4.接口和抽象类的区别是什么？



5.什么是值传递和引用传递？

值传递：实际参数把它的值传递给对应的形式参数，形式参数只是用实际参数的值初始化自己的存储单元内容，是两个不同的存储单元，方法执行中形式参数值的改变不影响实际参数的值。

引用传递：也称为传地址，方法调用时，实际参数是对象（或数字），这是实际参数与形式参数指向同一个地址，在方法执行中，对形式参数的操作实际上就是对实际参数的操作，这个结果在方法结束后被保留下来，方法执行中形式参数的改变将会影响实际参数。

6.Java中的两种异常类型是什么？他们有什么区别？

受检查的异常（checked exception）：从java.lang.Exception类衍生出来。定义方法时必须声明所有可能会抛出的checked exception，在调用这个方法时必须捕获它的checked exception，不然就得把它的exception传递下去。

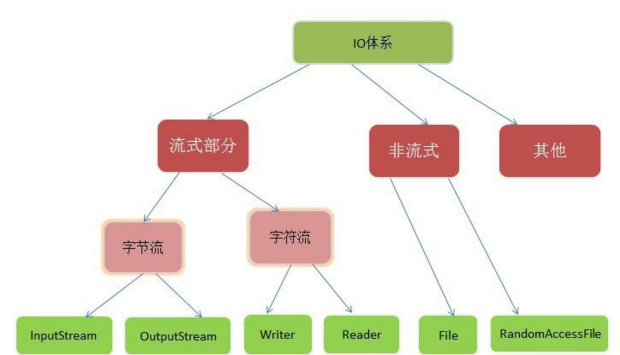
运行时异常（runtime exception）：从java.lang.RuntimeException或java.lang.Error类衍生出来。在定义方法时不需要声明会抛出runtime exception，在调用这个方法时不需要捕获这个runtime exception。

7.ArrayList和LinkedList有什么区别？

ArrayList：实现了List接口，它是以数组的方式来实现的，数组的特性是可以使用索引的方式来快速定位对象的位置，因此对于快速的随机取得对象的需求，使用ArrayList实现执行效率上会比较好。在删除可插入对象的动作时，ArrayList效率较低。

LinkedList：采用链表的方式来实现List接口的，它本身有自己特定的方法，如：addFirst()，addLast()，getFirst()，removeFirst()等。由于是采用链表实现的，因此在进行insert和remove动作时在效率上要比ArrayList要好得多！适合用来实现Stack(堆栈)与Queue(队列)，前者先进后出，后者是先进先出。

8.Java I/O的体系结构是什么？



其中其他类指文件读取部分的与安全相关的类,以及与本地操作系统相关的文件系统的类。

9.如何确保N各线程可以访问N个资源同时又不导致死锁？

指定获取锁的顺序，并强制线程按照指定的顺序获取锁。

10.在Java中，对象什么时候可以被垃圾回收？

当对象对当前使用这个对象的应用程序变得不可触及的时候，这个对象就可以被回收了。