1. **Java中数据类型**

分成四类：

整形：byte、int、short、long

字符型：char

浮点型：float、double

布尔型：boolean

1. **说出ArrayList,Vector, LinkedList的存储性能和特性**

先说ArrayList和Vector

两者都继承了抽象类AbstractList，但是Vector是线程安全的，而ArrayList是非线程安全的

再说ArrayList和LinkedList的区别

ArrayList 是数组结构，所以定位很快，但是插入和删除很慢

LinkedList 是双向链表结构，所以插入和删除很快，但是定位很慢。

线程安全类：

如果一个类，其****方法都是有synchronized修饰的****，那么该类就叫做****线程安全的类****  
同一时间，只有一个线程能够进入 ****这种类的一个实例**** 的去修改数据，进而保证了这个实例中的数据的安全(不会同时被多线程修改而变成脏数据)  
比如StringBuffer和StringBuilder的区别  
StringBuffer的方法都是有synchronized修饰的，StringBuffer就叫做线程安全的类  
而StringBuilder就不是线程安全的类。

1. **Collection和collections的区别：**

Collection是接口，是List和Set的父接口

Collections是工具类，提供了排序，混淆等等很多实用方法

1. HashMap和Hashtable的区别

HashMap和Hashtable都实现了Map接口，都是键**值对保存数据**的方式

区别1：

HashMap可以存放 null

Hashtable不能存放null

区别2：

HashMap不是线程安全的类

Hashtable是线程安全的类

1. final, finally, finalize的区别

final

final修饰类，方法，基本类型变量，引用的时候分别有不同的意思

修饰类 表示该类不能被继承

修饰方法 表示该方法不能被重写

修饰基本类型变量 表示该变量只能被赋值一次

修饰引用 表示该引用只有一次指向对象的机会

finally

finally 是用于异常处理的场面，无论是否有异常抛出，都会执行

finalize

finalize是Object的方法，所有类都继承了该方法。 当一个对象满足垃圾回收的条件，并且被回收的时候，其finalize()方法就会被调用

1. Overload和Override的区别，即重载和重写的区别。 Overloaded的方法是否可以改变返回值的类型?

Overload是方法重载的意思，指的是在同一个类里面，方法名一样，但是参数不一样

Override是方法重写的意思，指的是子类继承了父类的某个方法后，重新又写了一遍

Overloaded的方法是否可以改变返回值的类型?

**可以，重载其实本质上就是完全不同的方法，只是恰好取了相同的名字**

1. **Throw、throws的区别。**

Throws出现在方法声明上，throw出现在方法体内。

Throws指方法有可能出现一种异常，不一定出现，但是throw则是抛出了异常，执行throw一定抛出了某个对象异常。