# 集合的区别

## 1、Vector & ArrayList 的主要区别

1） 同步性:Vector是线程安全的，也就是说是同步的 ，而ArrayList 是线程序不安全的，不是同步的。

2）数据增长:当需要增长时,Vector默认增长为原来一倍 ，而ArrayList却是原来的50% ，这样,ArrayList就有利于节约内存空间。

如果涉及到堆栈，队列等操作，应该考虑用Vector，如果需要快速随机访问元素，应该使用ArrayList 。

扩展知识：

## 2. Hashtable & HashMap

Hashtable和HashMap它们的性能方面的比较类似 Vector和ArrayList，比如Hashtable的方法是同步的,而HashMap的不是。

## 3. ArrayList & LinkedList

ArrayList的内部实现是基于内部数组Object[],所以从概念上讲,它更象数组，但LinkedList的内部实现是基于一组连接的记录，所以，它更象一个链表结构，所以，它们在性能上有很大的差别：

从上面的分析可知,在ArrayList的前面或中间插入数据时,你必须将其后的所有数据相应的后移,这样必然要花费较多时间，所以,当你的操作是在一列数据的后面添加数据而不是在前面或中间,并且需要随机地访问其中的元素时,使用ArrayList会提供比较好的性能； 而访问链表中的某个元素时,就必须从链表的一端开始沿着连接方向一个一个元素地去查找,直到找到所需的元素为止，所以,当你的操作是在一列数据的前面或中间添加或删除数据，并且按照顺序访问其中的元素时，就应该使用LinkedList了。

# JSP内置对象有：

### 1.            输出输入对象:request对象、response对象、out对象

### 2.            通信控制对象:pageContext对象、session对象、application对象

### 3.            Servlet对象:page对象、config对象

### 4.            错误处理对象:exception对象

## 1.request对象

客户端的请求信息被封装在request对象中，通过它才能了解到客户的需求，然后做出响应。它是HttpServletRequest类的实例。

## 2.response对象

response对象包含了响应客户请求的有关信息，但在JSP中很少直接用到它。它是HttpServletResponse类的实例。

## 3.session对象

session对象指的是客户端与服务器的一次会话，从客户连到服务器的一个WebApplication开始，直到客户端与服务器断开连接为止。它是HttpSession类的实例.

## 4.out对象

out对象是JspWriter类的实例,是向客户端输出内容常用的对象

## 5.page对象

page对象就是指向当前JSP页面本身，有点象类中的this指针，它是java.lang.Object类的实例

## 6.application对象

application对象实现了用户间数据的共享，可存放全局变量。它开始于服务器的启动，直到服务器的关闭，在此期间，此对象将一直存在；这样在用户的前后连接或不同用户之间的连接中，可以对此对象的同一属性进行操作；在任何地方对此对象属性的操作，都将影响到其他用户对此的访问。服务器的启动和关闭决定了application对象的生命。它是ServletContext类的实例。

## 7.exception对象

exception对象是一个例外对象，当一个页面在运行过程中发生了例外，就产生这个对象。如果一个JSP页面要应用此对象，就必须把isErrorPage设为true，否则无法编译。他实际上是java.lang.Throwable的对象

## 8.pageContext对象

pageContext对象提供了对JSP页面内所有的对象及名字空间的访问，也就是说他可以访问到本页所在的SESSION，也可以取本页面所在的application的某一属性值，他相当于页面中所有功能的集大成者，它的本 类名也叫pageContext。

## 9.config对象

config对象是在一个Servlet初始化时，JSP引擎向它传递信息用的，此信息包括Servlet初始化时所要用到的参数（通过属性名和属性值构成）以及服务器的有关信息（通过传递一个ServletContext对象）