

ArrayListVectorLinkedList区别与用法Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/644/2021_2022_ArrayListV_c104_644882.htm

ArrayList 和 Vector 是采用数组方式存储数据，此数组元素数大于实际存储的数据以便增加和插入元素，都允许直接序号索引元素，但是插入数据要设计到数组元素移动等内存操作，所以索引数据快插入数据慢，Vector 由于使用了 synchronized 方法（线程安全）所以性能上比 ArrayList 要差，LinkedList 使用双向链表实现存储，按序号索引数据需要进行向前或向后遍历，但是插入数据时只需要记录本项的前后项即可，所以插入速度较快！线性表，链表，哈希表是常用的数据结构，在进行 Java 开发时，JDK 已经为我们提供了一系列相应的类来实现基本的数据结构。这些类均在 java.util 包中。本文试图通过简单的描述，向读者阐述各个类的作用以及如何正确使用这些类。Collection List LinkedList

ArrayList Vector Stack Set Map Hashtable

HashMap WeakHashMap Collection 接口 Collection 是最基本的集合接口，一个 Collection 代表一组 Object，即 Collection 的元素（Elements）。一些 Collection 允许相同的元素而另一些不行。一些能排序而另一些不行。Java SDK 不提供直接继承自 Collection 的类，Java SDK 提供的类都是继承自 Collection 的“子接口”如 List 和 Set。所有实现 Collection 接口的类都必须提供两个标准的构造函数：无参数的构造函数用于创建一个空的 Collection，有一个 Collection 参数的构造函数用于创建一个新的 Collection，这个新的 Collection 与传入的 Collection 有相同的元素。后一个构造函数允许用户复制一个 Collection。

如何遍历Collection中的每一个元素？不论Collection的实际类型如何，它都支持一个iterator()的方法，该方法返回一个迭代子，使用该迭代子即可逐一访问Collection中每一个元素。典型的用法如下：
Iterator it = collection.iterator(). // 获得一个迭代子
while(it.hasNext()) { Object obj = it.next(). // 得到下一个元素 }
由Collection接口派生的两个接口是List和Set。 100Test 下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com