

引入：

一、Collections

二、Colections工具类的常用操作方法

1、排序操作：（均为 static 方法）

2、查找、替换

3、同步控制

引入：

Collection和collections的区别？

Collection是存储一个一个元素的单列集合，主要实现接口是有List和Set接口。

Colections是一个操作Set、List和Map等集合的工具类

一、Collections

Collections 是一个操作 Set、List 和 Map 等集合的工具类

操作数组的工具类：Arrays

二、Colections工具类的常用操作方法

Collections 中提供了一系列静态的方法对集合元素进行排序、查询和修改等操作，还提供了对集合对象设置不可变、对集合对象实现同步控制等方法

1、排序操作：（均为 static 方法）

- 1 `reverse(List)`: 反转List中元素的顺序
- 2 `shuffle(List)`: 对List集合元素进行随机排序
- 3 `sort(List)`: 根据元素的自然顺序对指定的List集合元素按照升序排序
- 4 `sort(List, Comparator)`: 根据指定的Comparator产生的顺序对List集合元素进行排序
- 5 `swap(List, int, int)`: 将指定List集合中i处元素和j处元素进行交换

2、查找、替换

- 1 `Object max(Collection)`: 根据元素的自然顺序, 返回给定集合中的最大元素
- 2 `Object max(Collection, Comparator)`: 根据Comparator指定的顺序, 返回给定集合中元素的最大值

- 3 `Object min(Collection)`: 根据元素的自然顺序, 返回给定集合中的最小元素
- 4 `Object min(Collection, Comparator)`: 根据Comparator指定的顺序, 返回给定集合中元素的最小值
- 5 `int frequency(Collection, Object)`: 返回指定集合中指定元素的出现次数
- 6 `void copy(List dest, List src)`: 将src中的内容复制到dest中
- 7 `boolean replaceAll(List list, Object oldVal, Object newVal)`: 使用新值替换List对象的所有旧值

3、同步控制

Collections 类中提供了多个 `synchronizedXxx ()` 方法, 该方法可使将指定集合包装成线程同步的集合, 从而可以解决多线程并发访问集合时的线程安全问题

<code>static <T> Collection<T></code>	<code>synchronizedCollection(Collection<T> c)</code> Returns a synchronized (thread-safe) collection backed by the specified collection.
<code>static <T> List<T></code>	<code>synchronizedList(List<T> list)</code> Returns a synchronized (thread-safe) list backed by the specified list.
<code>static <K, V> Map<K, V></code>	<code>synchronizedMap(Map<K, V> m)</code> Returns a synchronized (thread-safe) map backed by the specified map.
<code>static <T> Set<T></code>	<code>synchronizedSet(Set<T> s)</code> Returns a synchronized (thread-safe) set backed by the specified set.
<code>static <K, V> SortedMap<K, V></code>	<code>synchronizedSortedMap(SortedMap<K, V> m)</code> Returns a synchronized (thread-safe) sorted map backed by the specified sorted map.
<code>static <T> SortedSet<T></code>	<code>synchronizedSortedSet(SortedSet<T> s)</code> Returns a synchronized (thread-safe) sorted set backed by the specified sorted set.