**Java学习第17天**

**知识点复习：注意 在使用API中已经定义的带泛型的类时，如果自己不加泛型那么会产生警报**

**1.集合工具类：Collections 对应01 ，02 03 对应11**

需求：对列表中的字符串元素做一个自然排序和一个长度排序。 也就是做一个练习， 需要做到查阅API做相关的练习。

其中有一个sort，还有查找的方法为binarySearch()这个方法，

用折半查找实现这个方法，折半查找的回顾,

2.替换反转API的练习 对应12

掌握一下集合中的fill、replaceAll和reverse这些有关替换和查找的API。

3.Collections.reverse 工具类 对应13

4.Collections reverseOrder 对应14

5.集合(Arrays) 对应15,16

几个常用的方法： asList

asList(T... a)

返回由指定数组支持的一个固定大小的列表。

toString(int[] a) 直接用就不用定义一个容器然后转换这么麻烦的操作了

返回指定数组的内容的字符串表示形式。

将数组变成List集合有什么好处？ 可以利用集合的思想与方法来操作数组，

但是注意一点：不能对集合中的元素增删，因为数组的长度是固定的。否则会发生：Exception in thread "main" java.lang.UnsupportedOperationException 这一个报错的方式。

能用:contains, get ,indexOf, subList();

如果数组中的元素都是对象，那么变成集合的时候，该数组中的元素就直接变成了集合中的元素。(可以使用基本类型封装类封装成对象)。

如果数组中的元素都是基本数据类型，那么变成集合的时候，该数组本身就会变成集合中的元素

1. 集合（集合转成数组） 对应17

集合转数组 Collections 中的方法 toArray

Object[] toArray()

返回一个数组，包含在这个列表中的所有元素在适当的顺序（从第一个到最后一个元素）。

<T> T[] toArray(T[] a)

返回一个数组，包含在这个列表中的所有元素在适当的顺序（从第一到最后一个元素）；返回数组的运行时类型是指定的数组的运行时类型。

将集合中的元素转为数组用的是Collections工具类中的toArray方法，

（将数组转为集合用的是Arrays工具类中的aslist()方法）。

如果数组的长度过小，就会创建新的数组存储;反之，如果新创建的数组的长度过大，那么剩余多的数组位置就会用null来补充。

所以，最好的方法就是创建一个正好大小的数组空间来存储。

7.集合(增强for循环) 对应18

用一种高级遍历的方法 for(数据类型 变量名：被遍历的的集合(Collection)或者数组)

这种高级遍历方法底层调用的是迭代器的遍历方法

第一个是ArrayList的数组

对集合进行遍历。

只能获取集合元素，但是不能对集合进行操作。

迭代器除了遍历，还可以进行remove集合中元素的动作。

如果是用ListIterator，还可以在遍历过程中进行增删改查的动作。

创建一个Hashmap<> 分别用keySet()和entrySet()的方法将它取出

\*/

// 思路：创建一个数组，然后用高级遍历方法将它输出。

8.集合（可变参数） 对应19

创建一个数组，扔进show方法中可以进一步进行操作

JDK1.5后面出现的新特性 可变参数 基本数据类型... 变量名 用三个点来表示 不用new数组传，直接传需要的参数

可变参数和python中的可变参数类似，python中是 \*参数名 一般可变参数写在最后面，而必要的参数是写在前面

1. StaticImport 静态导入 对应20

具体使用： 用import static 再导入具体的类名。

注意如果出现重名的函数或者方法那么就要写清楚具体的类名

// 这里编写的思路：创建一个int类型的数组，然后用sort方法排序，为二分查找做准备，然后用toString转为字符串输出。