Collator类 比较字符串（汉字大小）汉字排序

目录

[1. 本地化排序 1](#_Toc17720)

[2. 画数 1](#_Toc21991)

[3. Collator类 比较字符串（汉字大小）汉字排序 2](#_Toc8966)

[3.1. 自己动手拼音排序 3](#_Toc20180)

[4. ref 3](#_Toc8154)

# 本地化排序

前言：类 Collator

简介

Collator 类执行区分语言环境的 String 比较。使用此类可为自然语言文本构建搜索和排序例程。

Collator 是一个抽象基类。其子类实现具体的整理策略。Java 平台目前提供了 RuleBasedCollator 子类，它适用于很多种语言。还可以创建其他子类，以处理更多的专门需要。

与其他区分语言环境的类一样，可以使用静态工厂方法 getInstance 来为给定的语言环境获得适当的 Collator 对象。如果需要理解特定整理策略的细节或者需要修改策略，只需查看 Collator 的子类即可

---------------------

# 画数

// 升序 注意：是根据的汉字的拼音的字母排序的，而不是根据汉字一般的排序方法

**Collections**.*sort*(li, **Collator**.*getInstance*(**Locale**.***CHINA***));

# Collator类 比较字符串（汉字大小）汉字排序

Collator 类执行区分语言环境的 String 比较。使用此类可为自然语言文本构建搜索和排序例程。

Collator 是一个抽象基类。其子类实现具体的整理策略。Java 平台目前提供了 RuleBasedCollator 子类，它适用于很多种语言。还可以创建其他子类，以处理更多的专门需要。

与其他区分语言环境的类一样，可以使用静态工厂方法 getInstance 来为给定的语言环境获得适当的 Collator 对象。如果需要理解特定整理策略的细节或者需要修改策略，只需查看 Collator 的子类即可。

---------------------

 可以设置 Collator 的 strength 属性来确定比较中认为显著的差异级别。提供了四种 strength：PRIMARY、SECONDARY、TERTIARY 和 IDENTICAL。对语言特征的确切 strength 赋值和语言环境相关。例如在捷克语中，"e" 和 "f" 被认为是 PRIMARY 差异，而 "e" 和 "ě" 则是 SECONDARY 差异，"e" 和 "E" 是 TERTIARY 差异，"e" 和 "e" 是 IDENTICAL。下面的示例显示了如何针对美国英语忽略大小写和重音。

//Get the Collator for US English and set its strength to PRIMARY

Collator usCollator = Collator.getInstance(Locale.US);

usCollator.setStrength(Collator.PRIMARY);

if( usCollator.compare("abc", "ABC") == 0 ) {

System.out.println("Strings are equivalent");

---------------------

如果正好比较 String 一次，则 compare 方法可提供最佳性能。但在对 String 列表排序时，通常需要对每个 String 进行多次比较。在这种情况下，CollationKey 可提供更好的性能。CollationKey 类将一个 String 转换成一系列可与其他 CollationKey 进行按位比较的位。CollationKey 是由 Collator 对象为给定的 String 所创建的。

注：不能比较由不同 Collator 创建的 CollationKey。

Collator是一个抽象类，实现了Comparator和Clonable接口，Collator的构造方式有以下几种：函数生成一个RuleBasedCollator对象，此对象继承了Collator抽象类

## 自己动手拼音排序

之后我们需要实例化 Collator，并获取当前语言环境，通过重写比较器 Comparator来调用 Collections.sort() 方法，代码如下：

public static void sort() {

List<String> list = init();

Comparator<String> comparator = new Comparator<String>() {

public int compare(String o1, String o2) {

Collator collator = Collator.getInstance();

return collator.getCollationKey(o1).compareTo(

collator.getCollationKey(o2));

}

};

Collections.sort(list, comparator);

System.out.println(list);

}

---------------------

作者：JoJo小风

来源：CSDN

原文：https://blog.csdn.net/u013249965/article/details/52507343

版权声明：本文为博主原创文章，转载请附上博文链接！

# ref

---------------------

Collator类 比较字符串（汉字大小） - 侯春强的博客 - CSDN博客.html

Collator类详解（Collections.sort 中文排序） - JoJo小风 - CSDN博客.html