

warhin的专栏

目录视图

摘要视图

RSS 订阅

个人资料



warhin

访问：66030次
积分：584
等级：

BLOG 3

排名：千里之外

原创：22篇 转载：2篇
译文：0篇 评论：8条

文章搜索

文章分类

javascript (16)
java (4)
git (4)

文章存档

2016年08月 (2)
2016年07月 (2)
2016年04月 (2)
2016年03月 (8)
2015年07月 (4)

展开

阅读排行

Java Map按键排序和按值排序 (47294)
jedis客户端对象的使用 (3377)
DOM4J解析大数据的方法 (2314)
git revert远程常见操作 (2134)
git reset本地常见操作 (1520)
Struts2集成JSON的方案 (1445)
jQuery缓存探究 (1394)
jQuery transform类型实例 (784)
jQuery选择器探究：正则 (604)
jquery操作cookie (594)

【1024程序员节】我们的世界不只0和1 【观点】有了深度学习，你还学传统机器学习算法么？ 【知识库】深度学习知识图谱上线啦

Java Map按键排序和按值排序

标签： java string exception null 数据结构 jdk

2011-12-25 16:31 47359人阅读 评论(3) 收藏 举报

版权声明：本文为博主原创文章，未经博主允许不得转载。

目录(?) [+]

Map排序的方式有很多种，这里记录下自己总结的两种比较常用的方式：按键排序(sort by key)，按值排序(sort by value)。

按键排序(sort by key)

jdk内置的Java.util包下的TreeMap<K,V>既可满足此类需求，原理很简单，其重载的构造器之一

`TreeMap(Comparator<? super K> c)`
构造一个新的空映射，该映射根据给定的比较器进行排序。

有一个参数，该参数接受一个比较器，比较器定义比较规则，比较规则就是作用于TreeMap<K,V>的键，据此可实现按键排序。

```
[java]
01. public Map<String, String> sortMapByKey(Map<String, String> oriMap) {
02.     if (oriMap == null || oriMap.isEmpty()) {
03.         return null;
04.     }
05.     Map<String, String> sortedMap = new TreeMap<String, String>(new Comparator<String>() {
06.         public int compare(String key1, String key2) {
07.             int intKey1 = 0, intKey2 = 0;
08.             try {
09.                 intKey1 = getInt(key1);
10.                 intKey2 = getInt(key2);
11.             } catch (Exception e) {
12.                 intKey1 = 0;
13.                 intKey2 = 0;
14.             }
15.             return intKey1 - intKey2;
16.         });
17.     sortedMap.putAll(oriMap);
18.     return sortedMap;
19. }

20. private int getInt(String str) {
21.     int i = 0;
22.     try {
23.         Pattern p = Pattern.compile("^\\d+");
24.         Matcher m = p.matcher(str);
25.         if (m.find()) {
26.             i = Integer.valueOf(m.group());
27.         }
28.     } catch (NumberFormatException e) {
29.         e.printStackTrace();
30.     }
```

评论排行

- [Java Map按键排序和按值排序](#) (3)
- [jedis客户端对象的使用](#) (1)
- [DOM4J解析大数据的方案](#) (1)
- [jQuery异步框架探究1:jQuery](#) (1)
- [jquery操作cookie](#) (1)
- [jQuery transform类型实例](#) (1)
- [git push常见操作](#) (0)
- [jQuery缓存探究](#) (0)
- [git reset本地常见操作](#) (0)
- [git revert远程常见操作](#) (0)

推荐文章

- [* 2016 年最受欢迎的编程语言是什么？](#)
- [* Chromium扩展（Extension）的页面（Page）加载过程分析](#)
- [* Android Studio 2.2 来啦](#)
- [* 手把手教你做音乐播放器（二）技术原理与框架设计](#)
- [* JVM 性能调优实战之：使用阿里开源工具 TProfiler 在海量业务代码中精确定位性能代码](#)

最新评论

- [jQuery transform类型实例方法](#) 探吴士龙: 很细致呢, 感谢分享!
- [jQuery异步框架探究1:jQuery_D](#) 韩梦甜: 没有怎么用过, 先了解一下
- [Java Map按键排序和按值排序](#) 大愚若智_: 经常用得到, 收藏一把
- [Java Map按键排序和按值排序](#) a559927z: 不错不错不错不错不错不错不错不错
- [DOM4J解析大数据的方案](#) a363722188: 做个标记.....
- [Java Map按键排序和按值排序](#) 蒙奇_D_小彦: int value1 = 0, value2 = 0; ...
- [jquery操作cookie](#) SCOL_007: jQuery的扩展 很好很强大!
- [jedis客户端对象的使用](#) bjpowernode609: 哈哈 好~!



军用笔记本电脑



笔记本电脑排名



留学费用排名



呼叫中心

```
31.     }
32.     return i;
33. }
```

按值排序(sort by value)

按值排序就相对麻烦些了，貌似没有直接可用的数据结构能处理类似需求，需要我们自己转换一下。

Map本身按值排序是很有意义的，很多场合下都会遇到类似需求，可以认为其值是定义的某种规则或者权重。

```
[java]
01. public Map<String, String> sortMapByValue(Map<String, String> oriMap) {
02.     Map<String, String> sortedMap = new LinkedHashMap<String, String>();
03.     if (oriMap != null && !oriMap.isEmpty()) {
04.         List<Map.Entry<String, String>> entryList = new ArrayList<Map.Entry<String, String>>
(oriMap.entrySet());
05.         Collections.sort(entryList,
06.             new Comparator<Map.Entry<String, String>>() {
07.                 public int compare(Entry<String, String> entry1,
08.                     Entry<String, String> entry2) {
09.                     int value1 = 0, value2 = 0;
10.                     try {
11.                         value1 = getInt(entry1.getValue());
12.                         value2 = getInt(entry2.getValue());
13.                     } catch (NumberFormatException e) {
14.                         value1 = 0;
15.                         value2 = 0;
16.                     }
17.                     return value2 - value1;
18.                 }
19.             });
20.         Iterator<Map.Entry<String, String>> iter = entryList.iterator();
21.         Map.Entry<String, String> tmpEntry = null;
22.         while (iter.hasNext()) {
23.             tmpEntry = iter.next();
24.             sortedMap.put(tmpEntry.getKey(), tmpEntry.getValue());
25.         }
26.     }
27.     return sortedMap;
28. }
```

本例中先将待排序oriMap中的所有元素置于一个列表中，接着使用java.util.Collections的一个静态方法

<pre>static <T> void</pre>	<pre>sort(List<T> list, Comparator<? super T> c)</pre>
根据指定比较器产生的顺序对指定列表进行排序。	

来排序列表，同样是用比较器定义比较规则。排序后的列表中的元素再依次被装入Map，需要注意的一点是为了肯定的保证Map中元素与排序后的List中的元素的顺序一致，使用了LinkedHashMap数据类型，虽然该类型不常见，但是在一些特殊场合下还是非常有用的。

顶
7

踩
0

上一篇 [jedis客户端对象的使用](#)

下一篇 [jquery操作cookie](#)

[参考知识库](#)



算法与数据结构知识库
7778 关注 | 3123 收录



Java EE知识库
7292 关注 | 705 收录



Java SE知识库
14367 关注 | 459 收录



Java 知识库
16373 关注 | 1321 收录

猜你在找

- 数据结构 (JAVA版)
- 数据结构基础系列 (9)：排序
- 数据结构基础系列 (10)：外部排序
- 数据结构基础系列 (8)：查找
- 数据结构 (C版)
- Java Map按键排序和按值排序
- Java面试题如何对HashMap按键值排序
- JavaHashMap按键值排序
- 9Map按键和按键值排序
- Map按键排序sort by key 按值排序sort by value

聊天机器人 destination 苍月浏览器 美国高中留学 栉风沐雨 quartz
手机号码购买 139手机靓号 python 多线程 python多线程 剑桥历史 出国留学 intellij
南宋军队 靓号手机号码 环境与开发 军用笔记本电脑 深入的研究 手机号码贷款 西尔

查看评论

3楼 大愚若智_ 2016-03-25 11:21发表



经常用得到，收藏一把

2楼 a559927z 2016-02-26 11:03发表



不错不错不错不错不错不错不错

1楼 蒙奇_D_小彦 2013-01-28 17:53发表



```
int value1 = 0, value2 = 0;
try {
    value1 = getInt(entry1.getValue());
    value2 = getInt(entry2.getValue());
} catch (NumberFormatException e) {
    value1 = 0;
    value2 = 0;
}
中getInt(entry1.getValue())函数是什么对象的函数？
```

您还没有登录,请[登录](#)或[注册](#)

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

- 全部主题
- Hadoop AWS 移动游戏 Java Android iOS Swift 智能硬件 Docker OpenStack
- VPN Spark ERP IE10 Eclipse CRM JavaScript 数据库 Ubuntu NFC WAP jQuery
- BI HTML5 Spring Apache .NET API HTML SDK IIS Fedora XML LBS Unity
- Splashtop UML components Windows Mobile Rails QEMU KDE Cassandra CloudStack
- FTC coremail OPhone CouchBase 云计算 iOS6 Rackspace Web App SpringSide Maemo
- Compuware 大数据 aptech Perl Tornado Ruby Hibernate ThinkPHP HBase Pure Solr
- Angular Cloud Foundry Redis Scala Django Bootstrap

