

Externalizable和Serializable

原创 2015年12月03日 09:36:16 标签： Serializab / Externaliz / 序列化 / 反序列化 / transient 2646



0

1、Serializable序列化时不会调用默认的构造器，而Externalizable序列化时会调用默认构造器的！！！
2、Serializable：一个对象想要被序列化，那么它的类就要实现 此接口，这个对象的所有属性（包括private属性、包括其引用的对象）都可以被序列化和反序列化来保存、传递。
Externalizable：他是Serializable接口的子类，有时我们不希望序列化那么多，可以使用这个接口，这个接口的writeExternal()和readExternal()方法可以指定序列化哪些属性。

注意：
对象的序列化并不属于新的Reader和Writer层次结构的一部分，而是沿用老式的InputStream和OutputStream结构，在某些情况下，不得不混合使用两种类型的层次结构。

恢复了一个反序列化的对象后，如果想对其做更多的事情（对象.getClass().xxx），必须保证JVM能在本地类路径或者因特网的其他什么地方找到相关的.class文件。
恢复对象的默认构建器必须是public的，否则会抛异常。

由于Externalizable对象默认时不保存对象的任何字段，所以transient关键字只能伴随Serializable使用，虽然Externalizable对象中使用transient关键字也不报错，但不起任何作用。
3、把transient修饰的字段序列化。这也可以？of course！！

```
1 import java.io.*;
2 import java.util.*;
3
4 public class SerialCtl implements Serializable{
5     String a;
6     transient String b;
7     public SerialCtl(String aa, String bb){
8         a="Not Transient:"+aa;
9         b="Transient:"+bb;
10    }
11    public String toString(){
12        return a+"\n"+b;
13    }
14
15    private void writeObject(ObjectOutputStream o)throws IOException{
16        o.defaultWriteObject();
17        o.writeObject(b);
18    }
19    private void readObject(ObjectInputStream streamr)throws IOException, ClassNotFoundException
20    Exception{
21        streamr.defaultReadObject();
22        b=(String)streamr.readObject();
23    }
24
25    public static void main(String[] args){
26        SerialCtl sc =
27            new SerialCtl("Test1","Test2");
28        System.out.println("Before:\n"+sc);
29        ByteArrayOutputStream buf =
30            new ByteArrayOutputStream();
31        try{
32            ObjectOutputStream out1 = new
33                ObjectOutputStream(buf);
34
35            out1.writeObject(sc);
36
37            ObjectInputStream in1 = new
38                ObjectInputStream(new
39                    ByteArrayInputStream(buf.toByteArray()));
40
41            SerialCtl sc2 = (SerialCtl)in1.readObject();
42            System.out.println("After:\n"+sc2);
43
44        }catch(ClassNotFoundException e){
45            e.printStackTrace();
46        }catch(IOException e){
47            e.printStackTrace();
48
49        }
50    }
51 }
```

联系我们



请扫描二维码联系客服
webmaster@csdn.net
400-660-0108
QQ客服 客服论坛

关于 招聘 广告服务 百度
©1999-2018 CSDN版权所有
京ICP证09002463号

经营性网站备案信息
网络110报警服务
中国互联网举报中心
北京互联网违法和不良信息举报中心

博主最新文章 更多文章

ServerSocket 用法详解

Run.getRuntime.exec

java线程的状态

Thread的中断机制(interrupt)

ThreadPoolExecutor的corePoolSize和maximumPoolSize

文章分类

| | |
|--------|-----|
| JAVA开发 | 22篇 |
| C | 2篇 |
| 数据库 | 1篇 |

文章存档

| | |
|---------|----|
| 2016年9月 | 1篇 |
| 2016年8月 | 1篇 |
| 2016年6月 | 2篇 |
| 2016年5月 | 3篇 |
| 2016年3月 | 1篇 |
| 2016年2月 | 2篇 |

展开

博主热门文章

ServerSocket 用法详解

3720

Externalizable和Serializable

2607

msgrcv: Argument list too long

1408

GitBash 下如何解决javac/java编译运行乱码问题

648

java默认构造器constructor

588

Java 异常限制

452

Thread的中断机制(interrupt)

321

ThreadPoolExecutor的corePoolSize和maximumPoolSize

288

深入理解java中的try-catch-finally

260

Java中带包的类的编译与执行

👍

0

🔖

💬

```
Before:
Not Transient:Test1
Transient:Test2
After:
Not Transient:Test1
Transient:Test2

1  import java.io.*;
2  import java.util.*;
3
4  public class SerialCtl implements Serializable{
5      String a;
6      transient String b;
7      public SerialCtl(String aa, String bb){
8          a="Not Transient:"+aa;
9          b="Transient:"+bb;
10     }
11     public String toString(){
12         return a+"\n"+b;
13     }
14
15     private void writeObject(ObjectOutputStream o)throws IOException{
16         o.defaultWriteObject();
17         //      o.writeObject(b);
18     }
19     private void readObject(ObjectInputStream streamr)throws IOException, ClassNotFoundException
20     Exception{
21         streamr.defaultReadObject();
22         //      b=(String)streamr.readObject();
23     }
24
25     public static void main(String[] args){
26         SerialCtl sc =
27             new SerialCtl("Test1","Test2");
28         System.out.println("Before:\n"+sc);
29         ByteArrayOutputStream buf =
30             new ByteArrayOutputStream();
31         try{
32             ObjectOutputStream out1 = new
33                 ObjectOutputStream(buf);
34
35             out1.writeObject(sc);
36
37             ObjectInputStream in1 = new
38                 ObjectInputStream(new
39                     ByteArrayInputStream(buf.toByteArray()));
40
41             SerialCtl sc2 = (SerialCtl)in1.readObject();
42             System.out.println("After:\n"+sc2);
43
44             }catch(ClassNotFoundException e){
45                 e.printStackTrace();
46             }catch(IOException e){
47                 e.printStackTrace();
48
49             }
50         }
51     }
```

```
Before:
Not Transient:Test1
Transient:Test2
After:
Not Transient:Test1
null
```

结果：

方法writeObject处理对象的序列化。如果声明该方法，它将会被ObjectOutputStream调用而不是默认的序列化进程。如果你是第一次看见它，你会很惊奇尽管它们被外部类调用但事实上这是两个private的方法。并且它们既不存在于java.lang.Object，也没有在Serializable中声明。那么ObjectOutputStream如何使用它们的呢？这个吗，ObjectOutputStream使用了反射来寻找是否声明了这两个方法。因为ObjectOutputStream使用getPrivateMethod(通过getDeclareMethod可以获得私有方法)，所以这些方法不得不被声明为private以至于供ObjectOutputStream来使用。

在两个方法的开始处，你会发现调用了defaultWriteObject()和defaultReadObject()。它们做的是默认的序列化进程，就像写/读所有的non-transient和 non-static字段(但他们不会去做serialVersionUID的检查).通常说来，所有我们想要自己处理的字段都应该声明为transient。这样的话，defaultWriteObject/defaultReadObject便可以专注于其余字段，而我们则可为这些特定的字段(译者：指transient)定制序列化。使用那两个默认的方法并不是强

联系我们



请扫描二维码联系客服



webmaster@csdn.net



400-660-0108



QQ客服



客服论坛

关于 招聘 广告服务  百度

©1999-2018 CSDN版权所有

京ICP证09002463号

经营性网站备案信息

网络110报警服务

中国互联网举报中心

北京互联网违法和不良信息举报中心



查看更多>>

看 Python 如何诠释“薪”时代

Python全栈开发包含Python爬虫、前端、网站后台、Python机器学习与数据挖掘等, 从0基础小白到Python 企业级web开发达人、自动化运维开发能手的进击，课程真实企业项目实战演练，全面系统学习python编程语言，从容应对企业中各式各样的.....

29224

查看更多>>

👤 目前您尚未登录，请 [登录](#) 或 [注册](#) 后参与评论

JAVA 序列化 和 反序列化（ Externalizable Serializable ） 那些事

序列化控制 当我们对序列化进行控制时，可能某个特定子对象不想让Java序列化机制自动保存与恢复。如果子对象表示的是我们不希望将其序列化的敏感信息（如密码），通常会面临这种情况。即使对象中的这...

bigtree_3721 2016年01月13日 21:19 904

Externalizable 接口例子

Serializable接口声明的类的对象的内容都将被序列化，如果现在用户希望自己指定序列化的内容，则可以让一个类实现Externalizable接口，此接口定义如下： public inte...

Externalizable和Serializable

mengtuoling111 2015年12月03日 09:36 2646

1、Serializable序列化时不会调用默认构造器，而Externalizable序列化时会调用默认构造器的！！2、Serializable：一个对象想要被序列化，那么它的类就要实现 此接...

Java中的Externalizable接口

hirohrj123 2013年09月25日 10:16 1428

源码： package java.io; import java.io.ObjectOutput; import java.io.ObjectInput; public interface Ext...

Serializable和Externalizable

lovergo 2016年11月21日 14:07 908

无论是Serializable,还是Externalizable,他们都是实现序列化或者持久化的接口. 先解释下什么叫序列化,序列化就是对象保存到磁盘上,或者是对象由一台机器发送过来并保存下来就是序...

Serializable接口与Externalizable接口区别

lessonair 2013年12月22日 10:42 739

被Serializable接口声明的类的对象的内容都将被序列化，如果现在用户希望自己指定序列化的内容，则可以让一个类实现Externalizable接口，此接口定义如下： public inte...

java.io 序列化 总结Serializable 和 Externalizable 的区别与联系，以及输出后的乱码...

[java] view plain copy "WHITE-SPACE: pre"> 我们可以通过序列化来保存一个对象的状态（实例变量）到文件中，也可以从这个格式化的文件中很容易地读取对象的...

sdx1237 2017年03月02日 22:41 226

Java 序列化简单理解,serializable与externalizable区别？

什么是对象序列化？把Java对象状态保存为一组字节（序列化），之后还可以把这些字节组装成对象（反序列化）。为什么要使用对象序列化？1. 使用RMI（远程方法调用）2. 网络中传递对象（Ja...

love_world_ 2013年12月24日 21:46 1979

JAVA 对象序列化（三）——transient以及Externalizable的一种替代方法（使用Seri...

transient(瞬时)关键字 当我们队序列化进行控制时，可能某个特定子对象不想让Java的序列化机制自动保存与恢复。如果子对象表示的是我们不希望将其序列化的敏感信息（如密码），通...

djun100 2014年01月11日 17:49 1441

联系我们



请扫描二维码联系客服
✉ webmaster@csdn.net
☎ 400-660-0108
🗣 QQ客服 🗣 客服论坛

关于 招聘 广告服务 百度
©1999-2018 CSDN版权所有
京ICP证09002463号

经营性网站备案信息
网络110报警服务
中国互联网举报中心
北京互联网违法和不良信息举报中心

SERIALIZABLE与EXTERNALIZABLE两种序列化方式的区别

http://www.codingeek.com/java/io/differences-serializable-externalizable-interface-java-tutorial/

yyd19921214 2017年06月01日 17:54 201



17万欧起全家三代定居希腊！

全家三代享受免费教育与公立医疗，最快60天起即可定居希腊！



Java序列化之Serializable和Externalizable

a494303877 2016年11月28日 11:31 268

序列化：将数据内容流化，用于I/O读写，或者从磁盘读写等。 1、Serializable接口 正常情况下，只需要实现Serializable接口就能实现序列化。即便某些域是private修饰的，...

序列化--Serializable接口和Externalizable接口

zhai010101 2010年10月24日 19:47 297

序列化--Serializable接口和Externalizable接口序列化就是将对象转化为流的过程；反序列化就是把流中的数据转化为对象。要实现序列化与反序列化，则类必须实现Serializable...

Serializable和Externalizable浅析

VanishOrAlone 2017年12月18日 23:54 128

摘要: 本文主要要看看JDK中使用Serializable和Externalizable接口来完成Java对象序列化，并给出部分属性序列化的几种方式，最终做出Serializable和External...

序列化Serializable和Externalizable

u014714713 2017年03月16日 20:50 240

序列化就是将一个对象(标志对象的类型)及其状态转换为字节码，以文件，内存，数据库等形式保存起来。反序列化就是在适当的时候通过读取这些文件，得到原有状态的对象。 序列化可以通过实现两种接口来实现，即S...

Serializable和Externalizable

u011196335 2014年12月17日 23:52 267

1、Serializable自动序列化 public class Client1 { public static void main(String[] args) throws FileNotFo...

序列化：Serializable和Externalizable

xiefeifeihu 2012年08月29日 15:20 1888

Externalizable是深度定制序列化。如果同时实现了这两个接口，则只会执行Externalizable。 Serializable可以仅仅标记可序列化，使用JDK默认的序列化方...

Serializable 和 Externalizable

jumethh 2013年08月02日 18:11 571

1、序列化是干什么的？ 简单说就是为了保存在内存中的各种对象的状态（也就是实例变量，不是方法），并且可以把保存的对象状态再读出来。虽然你可以用你自己的各种各样的方法来保存object s...

Java序列化Serializable和Externalizable

guo_net 2014年08月11日 23:05 302

Java的对象序列化将那些实现了Serializable接口的对象转换chen

Serializable和Externalizable序列化反序列化

Smile_Luckly 2017年07月25日 15:01 161

原文地址：http://blog.csdn.net/u010233323/article/details/75253354 一、什么是序列化？ “对象序列化”（Object Serializa...

Java—序列化—Serializable—Externalizable

Ling912439122 2016年09月09日 16:00 569

众所周知，java支持对对象的序列化操作。为什么要序列化对象呢？总结起来也就一句话，对一个对象的持久化。也就是说该对象可以以某种形式保存起来，以便我在需要的时候对其反序列化操作后可以还原回来。 ...

联系我们



请扫描二维码联系客服

webmaster@csdn.net

400-660-0108

QQ客服 客服论坛

关于 招聘 广告服务 百度

©1999-2018 CSDN版权所有

京ICP证09002463号

经营性网站备案信息

网络110报警服务

中国互联网举报中心

北京互联网违法和不良信息举报中心