

JAVA序列化的两种方式计算机等级考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/645/2021_2022_JAVA_E5_BA_8F_E5_88_97_c97_645237.htm 大家都知道Serializable是一个

mark interface,告诉JVM这个对象可以被转换成二进制流来传输. Serializable 在我们实现这个接口的时候,我们可以使用4个私有方法来控制序列化的过程: 我们来看一个例子: public class FooImpl implements java.io.Serializable{ private String message. public String getFoo() { return message. } public void setMessage(String message) { this.message = message. } private void writeObject(java.io.ObjectOutputStream out) throws IOException { System.out.println("writeObject invoked").

out.writeObject(this.message == null ? "hohohahaha" : this.message). } private void readObject(java.io.ObjectInputStream in) throws IOException, ClassNotFoundException { System.out.println("readObject invoked"). this.message = (String) in.readObject(). } private Object writeReplace() throws ObjectOutputStreamException { System.out.println("writeReplace invoked"). return this. } private Object readResolve() throws ObjectOutputStreamException { System.out.println("readResolve invoked"). return this. } public Object serialize() throws IOException, ClassNotFoundException { ByteArrayOutputStream baos = new ByteArrayOutputStream(). ObjectOutputStream oos = new ObjectOutputStream(baos). oos.writeObject(this).

ByteArrayInputStream bais = new

ByteArrayInputStream(baos.toByteArray()). ObjectInputStream ois

```
= new ObjectInputStream(bais). return ois.readObject(). } public
static void main(String[] args) throws IOException,
ClassNotFoundException { FoolImpl fooimpl = new FoolImpl().
fooimpl.serialize(). } } public class FoolImpl implements
java.io.Serializable{ private String message. public String getFoo() {
return message. } public void setMessage(String message) {
this.message = message. } private void
writeObject(java.io.ObjectOutputStream out) throws IOException {
System.out.println("writeObject invoked").
out.writeObject(this.message == null ? "hohohahaha" :
this.message). } private void readObject(java.io.ObjectInputStream
in) throws IOException, ClassNotFoundException {
System.out.println("readObject invoked"). this.message = (String)
in.readObject(). } private Object writeReplace() throws
ObjectStreamException { System.out.println("writeReplace
invoked"). return this. } private Object readResolve() throws
ObjectStreamException { System.out.println("readResolve
invoked"). return this. } public Object serialize() throws
IOException, ClassNotFoundException { ByteArrayOutputStream
baos = new ByteArrayOutputStream(). ObjectOutputStream oos =
new ObjectOutputStream(baos). oos.writeObject(this).
ByteArrayInputStream bais = new
ByteArrayInputStream(baos.toByteArray()). ObjectInputStream ois
= new ObjectInputStream(bais). return ois.readObject(). } 100Test
下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com
```