

JAVA.IO.OBJECTOUTPUTSTREAM CLASS

http://www.tutorialspoint.com/java/io/java_io_objectoutputstream.htm

Copyright © tutorialspoint.com

Introduction

The **Java.io.ObjectOutputStream** class writes primitive data types and graphs of Java objects to an OutputStream. The objects can be read *reconstituted* using an ObjectInputStream.

Class declaration

Following is the declaration for **Java.io.ObjectOutputStream** class:

```
public class ObjectOutputStream
    extends OutputStream
    implements ObjectOutput, ObjectOutputStreamConstants
```

Class constructors

S.N. Constructor & Description

- protected ObjectOutputStream**
This provides a way for subclasses that are completely reimplementing ObjectOutputStream to not have to allocate private data just used by this implementation of ObjectOutputStream.
- ObjectOutputStreamOutputStreamout**
This creates an ObjectOutputStream that writes to the specified OutputStream.

Class methods

S.N. Method & Description

- [protected void annotateClassClass < ? > cl](#)
Subclasses may implement this method to allow class data to be stored in the stream.
- [protected void annotateProxyClassClass < ? > cl](#)
Subclasses may implement this method to store custom data in the stream along with descriptors for dynamic proxy classes.
- [void close](#)
This method closes the stream.
- [void defaultWriteObject](#)
This method writes the non-static and non-transient fields of the current class to this

stream.

5

[protected void drain](#)

This method drain any buffered data in ObjectOutputStream.

6

[protected boolean enableReplaceObjectbooleanenable](#)

This method enable the stream to do replacement of objects in the stream.

7

[void flush](#)

This method flushes the stream.

8

[ObjectOutputStream.PutField putFields](#)

This method retrieves the object used to buffer persistent fields to be written to the stream.

9

[protected Object replaceObjectObjectobj](#)

This method will allow trusted subclasses of ObjectOutputStream to substitute one object for another during serialization.

10

[void reset](#)

This method reset will disregard the state of any objects already written to the stream.

11

[void useProtocolVersionintversion](#)

This method specify stream protocol version to use when writing the stream.

12

[void writebyte\[\]buf](#)

This method writes an array of bytes.

13

[void writebyte\[\]buf, intoff, intlen](#)

This method writes a sub array of bytes.

14

[void writeintval](#)

This method writes a byte.

15

[void writeBooleanbooleanval](#)

This method writes a boolean.

16

[void writeByteintval](#)

This method writes an 8 bit byte.

17

[void writeBytesStringstr](#)

This method writes a String as a sequence of bytes.

18

[void writeCharintval](#)

This method writes a 16 bit char.

19

[void writeCharsStringstr](#)

This method writes a String as a sequence of chars.

20

[protected void writeClassDescriptorObjectStreamClassdesc](#)

This method writes the specified class descriptor to the ObjectOutputStream.

21

[void writeDoubledoubleval](#)

This method writes a 64 bit double.

22

[void writeFields](#)

This method writes the buffered fields to the stream.

23

[void writeFloatfloatval](#)

This method writes a 32 bit float.

24

[void writeIntintval](#)

This method writes a 32 bit int.

25

[void writeLonglongval](#)

This method writes a 64 bit long.

26

[void writeObjectObjectobj](#)

This method writes the specified object to the ObjectOutputStream.

27

[protected void writeObjectOverrideObjectobj](#)

This method is used by subclasses to override the default writeObject method.

28

[void writeShort*int*val](#)

This method writes a 16 bit short.

29

[protected void writeStreamHeader](#)

This method is provided so subclasses can append or prepend their own header to the stream.

30

[void writeUnsharedObject*obj*](#)

This method writes an "unshared" object to the ObjectOutputStream.

31

[void writerUTFString*str*](#)

This method primitive data write of this String in modified UTF-8 format.

Methods inherited

This class inherits methods from the following classes:

• java.io.Object

Loading [Mathjax]/jax/output/HTML-CSS/jax.js