| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DataOutput.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/io/DataInputStream.html)   [**NEXT CLASS**](http://docs.google.com/java/io/DataOutputStream.html) | [**FRAMES**](http://docs.google.com/index.html?java/io/DataOutput.html)    [**NO FRAMES**](http://docs.google.com/DataOutput.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **java.io**

Interface DataOutput

**All Known Subinterfaces:** [ImageOutputStream](http://docs.google.com/javax/imageio/stream/ImageOutputStream.html), [ObjectOutput](http://docs.google.com/java/io/ObjectOutput.html) **All Known Implementing Classes:** [DataOutputStream](http://docs.google.com/java/io/DataOutputStream.html), [FileCacheImageOutputStream](http://docs.google.com/javax/imageio/stream/FileCacheImageOutputStream.html), [FileImageOutputStream](http://docs.google.com/javax/imageio/stream/FileImageOutputStream.html), [ImageOutputStreamImpl](http://docs.google.com/javax/imageio/stream/ImageOutputStreamImpl.html), [MemoryCacheImageOutputStream](http://docs.google.com/javax/imageio/stream/MemoryCacheImageOutputStream.html), [ObjectOutputStream](http://docs.google.com/java/io/ObjectOutputStream.html), [RandomAccessFile](http://docs.google.com/java/io/RandomAccessFile.html)

public interface **DataOutput**

The DataOutput interface provides for converting data from any of the Java primitive types to a series of bytes and writing these bytes to a binary stream. There is also a facility for converting a String into [modified UTF-8](http://docs.google.com/DataInput.html#modified-utf-8) format and writing the resulting series of bytes.

For all the methods in this interface that write bytes, it is generally true that if a byte cannot be written for any reason, an IOException is thrown.

**Since:** JDK1.0 **See Also:**[DataInput](http://docs.google.com/java/io/DataInput.html), [DataOutputStream](http://docs.google.com/java/io/DataOutputStream.html)

| **Method Summary** | |
| --- | --- |
| void | [**write**](http://docs.google.com/java/io/DataOutput.html#write(byte%5B%5D))(byte[] b)            Writes to the output stream all the bytes in array b. |
| void | [**write**](http://docs.google.com/java/io/DataOutput.html#write(byte%5B%5D,%20int,%20int))(byte[] b, int off, int len)            Writes len bytes from array b, in order, to the output stream. |
| void | [**write**](http://docs.google.com/java/io/DataOutput.html#write(int))(int b)            Writes to the output stream the eight low-order bits of the argument b. |
| void | [**writeBoolean**](http://docs.google.com/java/io/DataOutput.html#writeBoolean(boolean))(boolean v)            Writes a boolean value to this output stream. |
| void | [**writeByte**](http://docs.google.com/java/io/DataOutput.html#writeByte(int))(int v)            Writes to the output stream the eight low- order bits of the argument v. |
| void | [**writeBytes**](http://docs.google.com/java/io/DataOutput.html#writeBytes(java.lang.String))([String](http://docs.google.com/java/lang/String.html) s)            Writes a string to the output stream. |
| void | [**writeChar**](http://docs.google.com/java/io/DataOutput.html#writeChar(int))(int v)            Writes a char value, which is comprised of two bytes, to the output stream. |
| void | [**writeChars**](http://docs.google.com/java/io/DataOutput.html#writeChars(java.lang.String))([String](http://docs.google.com/java/lang/String.html) s)            Writes every character in the string s, to the output stream, in order, two bytes per character. |
| void | [**writeDouble**](http://docs.google.com/java/io/DataOutput.html#writeDouble(double))(double v)            Writes a double value, which is comprised of eight bytes, to the output stream. |
| void | [**writeFloat**](http://docs.google.com/java/io/DataOutput.html#writeFloat(float))(float v)            Writes a float value, which is comprised of four bytes, to the output stream. |
| void | [**writeInt**](http://docs.google.com/java/io/DataOutput.html#writeInt(int))(int v)            Writes an int value, which is comprised of four bytes, to the output stream. |
| void | [**writeLong**](http://docs.google.com/java/io/DataOutput.html#writeLong(long))(long v)            Writes a long value, which is comprised of eight bytes, to the output stream. |
| void | [**writeShort**](http://docs.google.com/java/io/DataOutput.html#writeShort(int))(int v)            Writes two bytes to the output stream to represent the value of the argument. |
| void | [**writeUTF**](http://docs.google.com/java/io/DataOutput.html#writeUTF(java.lang.String))([String](http://docs.google.com/java/lang/String.html) s)            Writes two bytes of length information to the output stream, followed by the [modified UTF-8](http://docs.google.com/DataInput.html#modified-utf-8) representation of every character in the string s. |

| **Method Detail** |
| --- |

### write

void **write**(int b)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes to the output stream the eight low-order bits of the argument b. The 24 high-order bits of b are ignored.

**Parameters:**b - the byte to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### write

void **write**(byte[] b)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes to the output stream all the bytes in array b. If b is null, a NullPointerException is thrown. If b.length is zero, then no bytes are written. Otherwise, the byte b[0] is written first, then b[1], and so on; the last byte written is b[b.length-1].

**Parameters:**b - the data. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### write

void **write**(byte[] b,  
 int off,  
 int len)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes len bytes from array b, in order, to the output stream. If b is null, a NullPointerException is thrown. If off is negative, or len is negative, or off+len is greater than the length of the array b, then an IndexOutOfBoundsException is thrown. If len is zero, then no bytes are written. Otherwise, the byte b[off] is written first, then b[off+1], and so on; the last byte written is b[off+len-1].

**Parameters:**b - the data.off - the start offset in the data.len - the number of bytes to write. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### writeBoolean

void **writeBoolean**(boolean v)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes a boolean value to this output stream. If the argument v is true, the value (byte)1 is written; if v is false, the value (byte)0 is written. The byte written by this method may be read by the readBoolean method of interface DataInput, which will then return a boolean equal to v.

**Parameters:**v - the boolean to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### writeByte

void **writeByte**(int v)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes to the output stream the eight low- order bits of the argument v. The 24 high-order bits of v are ignored. (This means that writeByte does exactly the same thing as write for an integer argument.) The byte written by this method may be read by the readByte method of interface DataInput, which will then return a byte equal to (byte)v.

**Parameters:**v - the byte value to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### writeShort

void **writeShort**(int v)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes two bytes to the output stream to represent the value of the argument. The byte values to be written, in the order shown, are:

(byte)(0xff & (v >> 8))  
 (byte)(0xff & v)

The bytes written by this method may be read by the readShort method of interface DataInput , which will then return a short equal to (short)v.

**Parameters:**v - the short value to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### writeChar

void **writeChar**(int v)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes a char value, which is comprised of two bytes, to the output stream. The byte values to be written, in the order shown, are:

(byte)(0xff & (v >> 8))  
 (byte)(0xff & v)

The bytes written by this method may be read by the readChar method of interface DataInput , which will then return a char equal to (char)v.

**Parameters:**v - the char value to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### writeInt

void **writeInt**(int v)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes an int value, which is comprised of four bytes, to the output stream. The byte values to be written, in the order shown, are:

(byte)(0xff & (v >> 24))  
 (byte)(0xff & (v >> 16))  
 (byte)(0xff & (v >> 8))  
 (byte)(0xff & v)

The bytes written by this method may be read by the readInt method of interface DataInput , which will then return an int equal to v.

**Parameters:**v - the int value to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### writeLong

void **writeLong**(long v)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes a long value, which is comprised of eight bytes, to the output stream. The byte values to be written, in the order shown, are:

(byte)(0xff & (v >> 56))  
 (byte)(0xff & (v >> 48))  
 (byte)(0xff & (v >> 40))  
 (byte)(0xff & (v >> 32))  
 (byte)(0xff & (v >> 24))  
 (byte)(0xff & (v >> 16))  
 (byte)(0xff & (v >> 8))  
 (byte)(0xff & v)

The bytes written by this method may be read by the readLong method of interface DataInput , which will then return a long equal to v.

**Parameters:**v - the long value to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### writeFloat

void **writeFloat**(float v)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes a float value, which is comprised of four bytes, to the output stream. It does this as if it first converts this float value to an int in exactly the manner of the Float.floatToIntBits method and then writes the int value in exactly the manner of the writeInt method. The bytes written by this method may be read by the readFloat method of interface DataInput, which will then return a float equal to v.

**Parameters:**v - the float value to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### writeDouble

void **writeDouble**(double v)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes a double value, which is comprised of eight bytes, to the output stream. It does this as if it first converts this double value to a long in exactly the manner of the Double.doubleToLongBits method and then writes the long value in exactly the manner of the writeLong method. The bytes written by this method may be read by the readDouble method of interface DataInput, which will then return a double equal to v.

**Parameters:**v - the double value to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### writeBytes

void **writeBytes**([String](http://docs.google.com/java/lang/String.html) s)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes a string to the output stream. For every character in the string s, taken in order, one byte is written to the output stream. If s is null, a NullPointerException is thrown.

If s.length is zero, then no bytes are written. Otherwise, the character s[0] is written first, then s[1], and so on; the last character written is s[s.length-1]. For each character, one byte is written, the low-order byte, in exactly the manner of the writeByte method . The high-order eight bits of each character in the string are ignored.

**Parameters:**s - the string of bytes to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### writeChars

void **writeChars**([String](http://docs.google.com/java/lang/String.html) s)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes every character in the string s, to the output stream, in order, two bytes per character. If s is null, a NullPointerException is thrown. If s.length is zero, then no characters are written. Otherwise, the character s[0] is written first, then s[1], and so on; the last character written is s[s.length-1]. For each character, two bytes are actually written, high-order byte first, in exactly the manner of the writeChar method.

**Parameters:**s - the string value to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

### writeUTF

void **writeUTF**([String](http://docs.google.com/java/lang/String.html) s)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes two bytes of length information to the output stream, followed by the [modified UTF-8](http://docs.google.com/DataInput.html#modified-utf-8) representation of every character in the string s. If s is null, a NullPointerException is thrown. Each character in the string s is converted to a group of one, two, or three bytes, depending on the value of the character.

If a character c is in the range \u0001 through \u007f, it is represented by one byte:

(byte)c

If a character c is \u0000 or is in the range \u0080 through \u07ff, then it is represented by two bytes, to be written in the order shown:

(byte)(0xc0 | (0x1f & (c >> 6)))  
 (byte)(0x80 | (0x3f & c))

If a character c is in the range \u0800 through uffff, then it is represented by three bytes, to be written in the order shown:

(byte)(0xe0 | (0x0f & (c >> 12)))  
 (byte)(0x80 | (0x3f & (c >> 6)))  
 (byte)(0x80 | (0x3f & c))

First, the total number of bytes needed to represent all the characters of s is calculated. If this number is larger than 65535, then a UTFDataFormatException is thrown. Otherwise, this length is written to the output stream in exactly the manner of the writeShort method; after this, the one-, two-, or three-byte representation of each character in the string s is written.

The bytes written by this method may be read by the readUTF method of interface DataInput , which will then return a String equal to s.

**Parameters:**s - the string value to be written. **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - if an I/O error occurs.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DataOutput.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/io/DataInputStream.html)   [**NEXT CLASS**](http://docs.google.com/java/io/DataOutputStream.html) | [**FRAMES**](http://docs.google.com/index.html?java/io/DataOutput.html)    [**NO FRAMES**](http://docs.google.com/DataOutput.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).