| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Blob.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/sql/BatchUpdateException.html)   [**NEXT CLASS**](http://docs.google.com/java/sql/CallableStatement.html) | [**FRAMES**](http://docs.google.com/index.html?java/sql/Blob.html)    [**NO FRAMES**](http://docs.google.com/Blob.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **java.sql**

Interface Blob

**All Known Implementing Classes:** [SerialBlob](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.html)

public interface **Blob**

The representation (mapping) in the JavaTM programming language of an SQL BLOB value. An SQL BLOB is a built-in type that stores a Binary Large Object as a column value in a row of a database table. By default drivers implement Blob using an SQL locator(BLOB), which means that a Blob object contains a logical pointer to the SQL BLOB data rather than the data itself. A Blob object is valid for the duration of the transaction in which is was created.

Methods in the interfaces [ResultSet](http://docs.google.com/java/sql/ResultSet.html), [CallableStatement](http://docs.google.com/java/sql/CallableStatement.html), and [PreparedStatement](http://docs.google.com/java/sql/PreparedStatement.html), such as getBlob and setBlob allow a programmer to access an SQL BLOB value. The Blob interface provides methods for getting the length of an SQL BLOB (Binary Large Object) value, for materializing a BLOB value on the client, and for determining the position of a pattern of bytes within a BLOB value. In addition, this interface has methods for updating a BLOB value.

All methods on the Blob interface must be fully implemented if the JDBC driver supports the data type.

**Since:** 1.2

| **Method Summary** | |
| --- | --- |
| void | [**free**](http://docs.google.com/java/sql/Blob.html#free())()            This method frees the Blob object and releases the resources that it holds. |
| [InputStream](http://docs.google.com/java/io/InputStream.html) | [**getBinaryStream**](http://docs.google.com/java/sql/Blob.html#getBinaryStream())()            Retrieves the BLOB value designated by this Blob instance as a stream. |
| [InputStream](http://docs.google.com/java/io/InputStream.html) | [**getBinaryStream**](http://docs.google.com/java/sql/Blob.html#getBinaryStream(long,%20long))(long pos, long length)            Returns an InputStream object that contains a partial Blob value, starting with the byte specified by pos, which is length bytes in length. |
| byte[] | [**getBytes**](http://docs.google.com/java/sql/Blob.html#getBytes(long,%20int))(long pos, int length)            Retrieves all or part of the BLOB value that this Blob object represents, as an array of bytes. |
| long | [**length**](http://docs.google.com/java/sql/Blob.html#length())()            Returns the number of bytes in the BLOB value designated by this Blob object. |
| long | [**position**](http://docs.google.com/java/sql/Blob.html#position(java.sql.Blob,%20long))([Blob](http://docs.google.com/java/sql/Blob.html) pattern, long start)            Retrieves the byte position in the BLOB value designated by this Blob object at which pattern begins. |
| long | [**position**](http://docs.google.com/java/sql/Blob.html#position(byte%5B%5D,%20long))(byte[] pattern, long start)            Retrieves the byte position at which the specified byte array pattern begins within the BLOB value that this Blob object represents. |
| [OutputStream](http://docs.google.com/java/io/OutputStream.html) | [**setBinaryStream**](http://docs.google.com/java/sql/Blob.html#setBinaryStream(long))(long pos)            Retrieves a stream that can be used to write to the BLOB value that this Blob object represents. |
| int | [**setBytes**](http://docs.google.com/java/sql/Blob.html#setBytes(long,%20byte%5B%5D))(long pos, byte[] bytes)            Writes the given array of bytes to the BLOB value that this Blob object represents, starting at position pos, and returns the number of bytes written. |
| int | [**setBytes**](http://docs.google.com/java/sql/Blob.html#setBytes(long,%20byte%5B%5D,%20int,%20int))(long pos, byte[] bytes, int offset, int len)            Writes all or part of the given byte array to the BLOB value that this Blob object represents and returns the number of bytes written. |
| void | [**truncate**](http://docs.google.com/java/sql/Blob.html#truncate(long))(long len)            Truncates the BLOB value that this Blob object represents to be len bytes in length. |

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

### length

long **length**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Returns the number of bytes in the BLOB value designated by this Blob object.

**Returns:**length of the BLOB in bytes **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the length of the BLOB [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### getBytes

byte[] **getBytes**(long pos,  
 int length)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves all or part of the BLOB value that this Blob object represents, as an array of bytes. This byte array contains up to length consecutive bytes starting at position pos.

**Parameters:**pos - the ordinal position of the first byte in the BLOB value to be extracted; the first byte is at position 1length - the number of consecutive bytes to be copied; the value for length must be 0 or greater **Returns:**a byte array containing up to length consecutive bytes from the BLOB value designated by this Blob object, starting with the byte at position pos **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the BLOB value; if pos is less than 1 or length is less than 0 [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2 **See Also:**[setBytes(long, byte[])](http://docs.google.com/java/sql/Blob.html#setBytes(long,%20byte%5B%5D))

### getBinaryStream

[InputStream](http://docs.google.com/java/io/InputStream.html) **getBinaryStream**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the BLOB value designated by this Blob instance as a stream.

**Returns:**a stream containing the BLOB data **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the BLOB value [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2 **See Also:**[setBinaryStream(long)](http://docs.google.com/java/sql/Blob.html#setBinaryStream(long))

### position

long **position**(byte[] pattern,  
 long start)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the byte position at which the specified byte array pattern begins within the BLOB value that this Blob object represents. The search for pattern begins at position start.

**Parameters:**pattern - the byte array for which to searchstart - the position at which to begin searching; the first position is 1 **Returns:**the position at which the pattern appears, else -1 **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the BLOB or if start is less than 1 [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### position

long **position**([Blob](http://docs.google.com/java/sql/Blob.html) pattern,  
 long start)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the byte position in the BLOB value designated by this Blob object at which pattern begins. The search begins at position start.

**Parameters:**pattern - the Blob object designating the BLOB value for which to searchstart - the position in the BLOB value at which to begin searching; the first position is 1 **Returns:**the position at which the pattern begins, else -1 **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the BLOB value or if start is less than 1 [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### setBytes

int **setBytes**(long pos,  
 byte[] bytes)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Writes the given array of bytes to the BLOB value that this Blob object represents, starting at position pos, and returns the number of bytes written. The array of bytes will overwrite the existing bytes in the Blob object starting at the position pos. If the end of the Blob value is reached while writing the array of bytes, then the length of the Blob value will be increased to accomodate the extra bytes.

**Note:** If the value specified for pos is greater then the length+1 of the BLOB value then the behavior is undefined. Some JDBC drivers may throw a SQLException while other drivers may support this operation.

**Parameters:**pos - the position in the BLOB object at which to start writing; the first position is 1bytes - the array of bytes to be written to the BLOB value that this Blob object represents **Returns:**the number of bytes written **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the BLOB value or if pos is less than 1 [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.4 **See Also:**[getBytes(long, int)](http://docs.google.com/java/sql/Blob.html#getBytes(long,%20int))

### setBytes

int **setBytes**(long pos,  
 byte[] bytes,  
 int offset,  
 int len)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Writes all or part of the given byte array to the BLOB value that this Blob object represents and returns the number of bytes written. Writing starts at position pos in the BLOB value; len bytes from the given byte array are written. The array of bytes will overwrite the existing bytes in the Blob object starting at the position pos. If the end of the Blob value is reached while writing the array of bytes, then the length of the Blob value will be increased to accomodate the extra bytes.

**Note:** If the value specified for pos is greater then the length+1 of the BLOB value then the behavior is undefined. Some JDBC drivers may throw a SQLException while other drivers may support this operation.

**Parameters:**pos - the position in the BLOB object at which to start writing; the first position is 1bytes - the array of bytes to be written to this BLOB objectoffset - the offset into the array bytes at which to start reading the bytes to be setlen - the number of bytes to be written to the BLOB value from the array of bytes bytes **Returns:**the number of bytes written **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the BLOB value or if pos is less than 1 [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.4 **See Also:**[getBytes(long, int)](http://docs.google.com/java/sql/Blob.html#getBytes(long,%20int))

### setBinaryStream

[OutputStream](http://docs.google.com/java/io/OutputStream.html) **setBinaryStream**(long pos)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves a stream that can be used to write to the BLOB value that this Blob object represents. The stream begins at position pos. The bytes written to the stream will overwrite the existing bytes in the Blob object starting at the position pos. If the end of the Blob value is reached while writing to the stream, then the length of the Blob value will be increased to accomodate the extra bytes.

**Note:** If the value specified for pos is greater then the length+1 of the BLOB value then the behavior is undefined. Some JDBC drivers may throw a SQLException while other drivers may support this operation.

**Parameters:**pos - the position in the BLOB value at which to start writing; the first position is 1 **Returns:**a java.io.OutputStream object to which data can be written **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the BLOB value or if pos is less than 1 [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.4 **See Also:**[getBinaryStream()](http://docs.google.com/java/sql/Blob.html#getBinaryStream())

### truncate

void **truncate**(long len)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Truncates the BLOB value that this Blob object represents to be len bytes in length.

**Note:** If the value specified for pos is greater then the length+1 of the BLOB value then the behavior is undefined. Some JDBC drivers may throw a SQLException while other drivers may support this operation.

**Parameters:**len - the length, in bytes, to which the BLOB value that this Blob object represents should be truncated **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the BLOB value or if len is less than 0 [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.4

### free

void **free**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

This method frees the Blob object and releases the resources that it holds. The object is invalid once the free method is called.

After free has been called, any attempt to invoke a method other than free will result in a SQLException being thrown. If free is called multiple times, the subsequent calls to free are treated as a no-op.

**Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs releasing the Blob's resources [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.6

### getBinaryStream

[InputStream](http://docs.google.com/java/io/InputStream.html) **getBinaryStream**(long pos,  
 long length)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Returns an InputStream object that contains a partial Blob value, starting with the byte specified by pos, which is length bytes in length.

**Parameters:**pos - the offset to the first byte of the partial value to be retrieved. The first byte in the Blob is at position 1length - the length in bytes of the partial value to be retrieved **Returns:**InputStream through which the partial Blob value can be read. **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if pos is less than 1 or if pos is greater than the number of bytes in the Blob or if pos + length is greater than the number of bytes in the Blob [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.6

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Blob.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/sql/BatchUpdateException.html)   [**NEXT CLASS**](http://docs.google.com/java/sql/CallableStatement.html) | [**FRAMES**](http://docs.google.com/index.html?java/sql/Blob.html)    [**NO FRAMES**](http://docs.google.com/Blob.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).