| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/SerialBlob.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/javax/sql/rowset/serial/SerialArray.html)   [**NEXT CLASS**](http://docs.google.com/javax/sql/rowset/serial/SerialClob.html) | [**FRAMES**](http://docs.google.com/index.html?javax/sql/rowset/serial/SerialBlob.html)    [**NO FRAMES**](http://docs.google.com/SerialBlob.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#2s8eyo1) |

## **javax.sql.rowset.serial**

Class SerialBlob

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **javax.sql.rowset.serial.SerialBlob**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [Cloneable](http://docs.google.com/java/lang/Cloneable.html), [Blob](http://docs.google.com/java/sql/Blob.html)

public class **SerialBlob**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Blob](http://docs.google.com/java/sql/Blob.html), [Serializable](http://docs.google.com/java/io/Serializable.html), [Cloneable](http://docs.google.com/java/lang/Cloneable.html)

A serialized mapping in the Java programming language of an SQL BLOB value.

The SerialBlob class provides a constructor for creating an instance from a Blob object. Note that the Blob object should have brought the SQL BLOB value's data over to the client before a SerialBlob object is constructed from it. The data of an SQL BLOB value can be materialized on the client as an array of bytes (using the method Blob.getBytes) or as a stream of uninterpreted bytes (using the method Blob.getBinaryStream).

SerialBlob methods make it possible to make a copy of a SerialBlob object as an array of bytes or as a stream. They also make it possible to locate a given pattern of bytes or a Blob object within a SerialBlob object and to update or truncate a Blob object.

**See Also:**[Serialized Form](http://docs.google.com/serialized-form.html#javax.sql.rowset.serial.SerialBlob)

| **Constructor Summary** | |
| --- | --- |
| [**SerialBlob**](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.html#SerialBlob(java.sql.Blob))([Blob](http://docs.google.com/java/sql/Blob.html) blob)            Constructs a SerialBlob object that is a serialized version of the given Blob object. |
| [**SerialBlob**](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.html#SerialBlob(byte%5B%5D))(byte[] b)            Constructs a SerialBlob object that is a serialized version of the given byte array. |

| **Method Summary** | |
| --- | --- |
| void | [**free**](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.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/javax/sql/rowset/serial/SerialBlob.html#getBinaryStream())()            Returns this SerialBlob object as an input stream. |
| [InputStream](http://docs.google.com/java/io/InputStream.html) | [**getBinaryStream**](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.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/javax/sql/rowset/serial/SerialBlob.html#getBytes(long,%20int))(long pos, int length)            Copies the specified number of bytes, starting at the given position, from this SerialBlob object to another array of bytes. |
| long | [**length**](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.html#length())()            Retrieves the number of bytes in this SerialBlob object's array of bytes. |
| long | [**position**](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.html#position(java.sql.Blob,%20long))([Blob](http://docs.google.com/java/sql/Blob.html) pattern, long start)            Returns the position in this SerialBlob object where the given Blob object begins, starting the search at the specified position. |
| long | [**position**](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.html#position(byte%5B%5D,%20long))(byte[] pattern, long start)            Returns the position in this SerialBlob object where the given pattern of bytes begins, starting the search at the specified position. |
| [OutputStream](http://docs.google.com/java/io/OutputStream.html) | [**setBinaryStream**](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.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/javax/sql/rowset/serial/SerialBlob.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/javax/sql/rowset/serial/SerialBlob.html#setBytes(long,%20byte%5B%5D,%20int,%20int))(long pos, byte[] bytes, int offset, int length)            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/javax/sql/rowset/serial/SerialBlob.html#truncate(long))(long length)            Truncates the BLOB value that this Blob object represents to be len bytes in length. |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Constructor Detail** |
| --- |

### SerialBlob

public **SerialBlob**(byte[] b)  
 throws [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html),  
 [SQLException](http://docs.google.com/java/sql/SQLException.html)

Constructs a SerialBlob object that is a serialized version of the given byte array.

The new SerialBlob object is initialized with the data from the byte array, thus allowing disconnected RowSet objects to establish serialized Blob objects without touching the data source.

**Parameters:**b - the byte array containing the data for the Blob object to be serialized **Throws:** [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html) - if an error occurs during serialization [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a SQL errors occurs

### SerialBlob

public **SerialBlob**([Blob](http://docs.google.com/java/sql/Blob.html) blob)  
 throws [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html),  
 [SQLException](http://docs.google.com/java/sql/SQLException.html)

Constructs a SerialBlob object that is a serialized version of the given Blob object.

The new SerialBlob object is initialized with the data from the Blob object; therefore, the Blob object should have previously brought the SQL BLOB value's data over to the client from the database. Otherwise, the new SerialBlob object will contain no data.

**Parameters:**blob - the Blob object from which this SerialBlob object is to be constructed; cannot be null. **Throws:** [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html) - if an error occurs during serialization [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the Blob passed to this to this constructor is a null.**See Also:**[Blob](http://docs.google.com/java/sql/Blob.html)

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

### getBytes

public byte[] **getBytes**(long pos,  
 int length)  
 throws [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html)

Copies the specified number of bytes, starting at the given position, from this SerialBlob object to another array of bytes.

Note that if the given number of bytes to be copied is larger than the length of this SerialBlob object's array of bytes, the given number will be shortened to the array's length.

**Specified by:**[getBytes](http://docs.google.com/java/sql/Blob.html#getBytes(long,%20int)) in interface [Blob](http://docs.google.com/java/sql/Blob.html) **Parameters:**pos - the ordinal position of the first byte in this SerialBlob object to be copied; numbering starts at 1; must not be less than 1 and must be less than or equal to the length of this SerialBlob objectlength - the number of bytes to be copied **Returns:**an array of bytes that is a copy of a region of this SerialBlob object, starting at the given position and containing the given number of consecutive bytes **Throws:** [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html) - if the given starting position is out of bounds**See Also:**[Blob.setBytes(long, byte[])](http://docs.google.com/java/sql/Blob.html#setBytes(long,%20byte%5B%5D))

### length

public long **length**()  
 throws [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html)

Retrieves the number of bytes in this SerialBlob object's array of bytes.

**Specified by:**[length](http://docs.google.com/java/sql/Blob.html#length()) in interface [Blob](http://docs.google.com/java/sql/Blob.html) **Returns:**a long indicating the length in bytes of this SerialBlob object's array of bytes **Throws:** [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html) - if an error occurs

### getBinaryStream

public [InputStream](http://docs.google.com/java/io/InputStream.html) **getBinaryStream**()  
 throws [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html)

Returns this SerialBlob object as an input stream. Unlike the related method, setBinaryStream, a stream is produced regardless of whether the SerialBlob was created with a Blob object or a byte array.

**Specified by:**[getBinaryStream](http://docs.google.com/java/sql/Blob.html#getBinaryStream()) in interface [Blob](http://docs.google.com/java/sql/Blob.html) **Returns:**a java.io.InputStream object that contains this SerialBlob object's array of bytes **Throws:** [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html) - if an error occurs**See Also:**[setBinaryStream(long)](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.html#setBinaryStream(long))

### position

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

Returns the position in this SerialBlob object where the given pattern of bytes begins, starting the search at the specified position.

**Specified by:**[position](http://docs.google.com/java/sql/Blob.html#position(byte%5B%5D,%20long)) in interface [Blob](http://docs.google.com/java/sql/Blob.html) **Parameters:**pattern - the pattern of bytes for which to searchstart - the position of the byte in this SerialBlob object from which to begin the search; the first position is 1; must not be less than 1 nor greater than the length of this SerialBlob object **Returns:**the position in this SerialBlob object where the given pattern begins, starting at the specified position; -1 if the pattern is not found or the given starting position is out of bounds; position numbering for the return value starts at 1 **Throws:** [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html) - if an error occurs when serializing the blob [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the BLOB value from the database

### position

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

Returns the position in this SerialBlob object where the given Blob object begins, starting the search at the specified position.

**Specified by:**[position](http://docs.google.com/java/sql/Blob.html#position(java.sql.Blob,%20long)) in interface [Blob](http://docs.google.com/java/sql/Blob.html) **Parameters:**pattern - the Blob object for which to search;start - the position of the byte in this SerialBlob object from which to begin the search; the first position is 1; must not be less than 1 nor greater than the length of this SerialBlob object **Returns:**the position in this SerialBlob object where the given Blob object begins, starting at the specified position; -1 if the pattern is not found or the given starting position is out of bounds; position numbering for the return value starts at 1 **Throws:** [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html) - if an error occurs when serializing the blob [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the BLOB value from the database

### setBytes

public int **setBytes**(long pos,  
 byte[] bytes)  
 throws [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html),  
 [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.

**Specified by:**[setBytes](http://docs.google.com/java/sql/Blob.html#setBytes(long,%20byte%5B%5D)) in interface [Blob](http://docs.google.com/java/sql/Blob.html) **Parameters:**pos - the position in the SQL BLOB value at which to start writing. The first position is 1; must not be less than 1 nor greater than the length of this SerialBlob object.bytes - the array of bytes to be written to the BLOB value that this Blob object represents **Returns:**the number of bytes written **Throws:** [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html) - if there is an error accessing the BLOB value; or if an invalid position is set; if an invalid offset value is set [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the BLOB value from the database**See Also:**[getBytes(long, int)](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.html#getBytes(long,%20int))

### setBytes

public int **setBytes**(long pos,  
 byte[] bytes,  
 int offset,  
 int length)  
 throws [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html),  
 [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.

**Specified by:**[setBytes](http://docs.google.com/java/sql/Blob.html#setBytes(long,%20byte%5B%5D,%20int,%20int)) in interface [Blob](http://docs.google.com/java/sql/Blob.html) **Parameters:**pos - the position in the BLOB object at which to start writing. The first position is 1; must not be less than 1 nor greater than the length of this SerialBlob object.bytes - the array of bytes to be written to the BLOB valueoffset - the offset in the byte array at which to start reading the bytes. The first offset position is 0; must not be less than 0 nor greater than the length of the byte arraylength - the number of bytes to be written to the BLOB value from the array of bytes *bytes*. **Returns:**the number of bytes written **Throws:** [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html) - if there is an error accessing the BLOB value; if an invalid position is set; if an invalid offset value is set; if number of bytes to be written is greater than the SerialBlob length; or the combined values of the length and offset is greater than the Blob buffer [SQLException](http://docs.google.com/java/sql/SQLException.html) - if there is an error accessing the BLOB value from the database.**See Also:**[getBytes(long, int)](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.html#getBytes(long,%20int))

### setBinaryStream

public [OutputStream](http://docs.google.com/java/io/OutputStream.html) **setBinaryStream**(long pos)  
 throws [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html),  
 [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. This method forwards the setBinaryStream() call to the underlying Blob in the event that this SerialBlob object is instantiated with a Blob. If this SerialBlob is instantiated with a byte array, a SerialException is thrown.

**Specified by:**[setBinaryStream](http://docs.google.com/java/sql/Blob.html#setBinaryStream(long)) in interface [Blob](http://docs.google.com/java/sql/Blob.html) **Parameters:**pos - the position in the BLOB value at which to start writing **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 [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html) - if the SerialBlob in not instantiated with a Blob object that supports setBinaryStream()**See Also:**[getBinaryStream()](http://docs.google.com/javax/sql/rowset/serial/SerialBlob.html#getBinaryStream())

### truncate

public void **truncate**(long length)  
 throws [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html)

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

**Specified by:**[truncate](http://docs.google.com/java/sql/Blob.html#truncate(long)) in interface [Blob](http://docs.google.com/java/sql/Blob.html) **Parameters:**length - the length, in bytes, to which the BLOB value that this Blob object represents should be truncated **Throws:** [SerialException](http://docs.google.com/javax/sql/rowset/serial/SerialException.html) - if there is an error accessing the Blob value; or the length to truncate is greater that the SerialBlob length

### getBinaryStream

public [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.

**Specified by:**[getBinaryStream](http://docs.google.com/java/sql/Blob.html#getBinaryStream(long,%20long)) in interface [Blob](http://docs.google.com/java/sql/Blob.html) **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**Since:** 1.6

### free

public 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. Blob object. The object is invalid once the free method is called. If free is called multiple times, the subsequent calls to free are treated as a no-op.

**Specified by:**[free](http://docs.google.com/java/sql/Blob.html#free()) in interface [Blob](http://docs.google.com/java/sql/Blob.html) **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs releasing the Blob's resources**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/SerialBlob.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/javax/sql/rowset/serial/SerialArray.html)   [**NEXT CLASS**](http://docs.google.com/javax/sql/rowset/serial/SerialClob.html) | [**FRAMES**](http://docs.google.com/index.html?javax/sql/rowset/serial/SerialBlob.html)    [**NO FRAMES**](http://docs.google.com/SerialBlob.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#2s8eyo1) |

[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).