| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/SQLInputImpl.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/SerialStruct.html)   [**NEXT CLASS**](http://docs.google.com/javax/sql/rowset/serial/SQLOutputImpl.html) | [**FRAMES**](http://docs.google.com/index.html?javax/sql/rowset/serial/SQLInputImpl.html)    [**NO FRAMES**](http://docs.google.com/SQLInputImpl.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

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

Class SQLInputImpl

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

**All Implemented Interfaces:** [SQLInput](http://docs.google.com/java/sql/SQLInput.html)

public class **SQLInputImpl**extends [Object](http://docs.google.com/java/lang/Object.html)implements [SQLInput](http://docs.google.com/java/sql/SQLInput.html)

An input stream used for custom mapping user-defined types (UDTs). An SQLInputImpl object is an input stream that contains a stream of values that are the attributes of a UDT.

This class is used by the driver behind the scenes when the method getObject is called on an SQL structured or distinct type that has a custom mapping; a programmer never invokes SQLInputImpl methods directly. They are provided here as a convenience for those who write RowSet implementations.

The SQLInputImpl class provides a set of reader methods analogous to the ResultSet getter methods. These methods make it possible to read the values in an SQLInputImpl object.

The method wasNull is used to determine whether the the last value read was SQL NULL.

When the method getObject is called with an object of a class implementing the interface SQLData, the JDBC driver calls the method SQLData.getSQLType to determine the SQL type of the UDT being custom mapped. The driver creates an instance of SQLInputImpl, populating it with the attributes of the UDT. The driver then passes the input stream to the method SQLData.readSQL, which in turn calls the SQLInputImpl reader methods to read the attributes from the input stream.

**See Also:**[SQLData](http://docs.google.com/java/sql/SQLData.html)

| **Constructor Summary** | |
| --- | --- |
| [**SQLInputImpl**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#SQLInputImpl(java.lang.Object%5B%5D,%20java.util.Map))([Object](http://docs.google.com/java/lang/Object.html)[] attributes, [Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),[Class](http://docs.google.com/java/lang/Class.html)<?>> map)            Creates an SQLInputImpl object initialized with the given array of attributes and the given type map. |

| **Method Summary** | |
| --- | --- |
| [Array](http://docs.google.com/java/sql/Array.html) | [**readArray**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readArray())()            Reads an SQL ARRAY value from the stream and returns it as an Array object in the Java programming language. |
| [InputStream](http://docs.google.com/java/io/InputStream.html) | [**readAsciiStream**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readAsciiStream())()            Returns the next attribute in this SQLInputImpl object as a stream of ASCII characters. |
| [BigDecimal](http://docs.google.com/java/math/BigDecimal.html) | [**readBigDecimal**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readBigDecimal())()            Retrieves the next attribute in this SQLInputImpl object as a java.math.BigDecimal. |
| [InputStream](http://docs.google.com/java/io/InputStream.html) | [**readBinaryStream**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readBinaryStream())()            Returns the next attribute in this SQLInputImpl object as a stream of uninterpreted bytes. |
| [Blob](http://docs.google.com/java/sql/Blob.html) | [**readBlob**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readBlob())()            Retrieves the BLOB value at the head of this SQLInputImpl object as a Blob object in the Java programming language. |
| boolean | [**readBoolean**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readBoolean())()            Retrieves the next attribute in this SQLInputImpl object as a boolean in the Java programming language. |
| byte | [**readByte**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readByte())()            Retrieves the next attribute in this SQLInputImpl object as a byte in the Java programming language. |
| byte[] | [**readBytes**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readBytes())()            Retrieves the next attribute in this SQLInputImpl object as an array of bytes. |
| [Reader](http://docs.google.com/java/io/Reader.html) | [**readCharacterStream**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readCharacterStream())()            Retrieves the next attribute in this SQLInputImpl object as a stream of Unicode characters. |
| [Clob](http://docs.google.com/java/sql/Clob.html) | [**readClob**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readClob())()            Retrieves the CLOB value at the head of this SQLInputImpl object as a Clob object in the Java programming language. |
| [Date](http://docs.google.com/java/sql/Date.html) | [**readDate**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readDate())()            Retrieves the next attribute in this SQLInputImpl as a java.sql.Date object. |
| double | [**readDouble**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readDouble())()            Retrieves the next attribute in this SQLInputImpl object as a double in the Java programming language. |
| float | [**readFloat**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readFloat())()            Retrieves the next attribute in this SQLInputImpl object as a float in the Java programming language. |
| int | [**readInt**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readInt())()            Retrieves the next attribute in this SQLInputImpl object as an int in the Java programming language. |
| long | [**readLong**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readLong())()            Retrieves the next attribute in this SQLInputImpl object as a long in the Java programming language. |
| [NClob](http://docs.google.com/java/sql/NClob.html) | [**readNClob**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readNClob())()            Reads an SQL NCLOB value from the stream and returns it as a Clob object in the Java programming language. |
| [String](http://docs.google.com/java/lang/String.html) | [**readNString**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readNString())()            Reads the next attribute in the stream and returns it as a String in the Java programming language. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**readObject**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readObject())()            Retrieves the value at the head of this SQLInputImpl object as an Object in the Java programming language. |
| [Ref](http://docs.google.com/java/sql/Ref.html) | [**readRef**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readRef())()            Retrieves the value at the head of this SQLInputImpl object as a Ref object in the Java programming language. |
| [RowId](http://docs.google.com/java/sql/RowId.html) | [**readRowId**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readRowId())()            Reads an SQL ROWID value from the stream and returns it as a RowId object in the Java programming language. |
| short | [**readShort**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readShort())()            Retrieves the next attribute in this SQLInputImpl object as a short in the Java programming language. |
| [SQLXML](http://docs.google.com/java/sql/SQLXML.html) | [**readSQLXML**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readSQLXML())()            Reads an SQL XML value from the stream and returns it as a SQLXML object in the Java programming language. |
| [String](http://docs.google.com/java/lang/String.html) | [**readString**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readString())()            Retrieves the next attribute in this SQLInputImpl object as a String in the Java programming language. |
| [Time](http://docs.google.com/java/sql/Time.html) | [**readTime**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readTime())()            Retrieves the next attribute in this SQLInputImpl object as a java.sql.Time object. |
| [Timestamp](http://docs.google.com/java/sql/Timestamp.html) | [**readTimestamp**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readTimestamp())()            Retrieves the next attribute in this SQLInputImpl object as a java.sql.Timestamp object. |
| [URL](http://docs.google.com/java/net/URL.html) | [**readURL**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#readURL())()            Reads an SQL DATALINK value from the stream and returns it as an URL object in the Java programming language. |
| boolean | [**wasNull**](http://docs.google.com/javax/sql/rowset/serial/SQLInputImpl.html#wasNull())()            Ascertains whether the last value read from this SQLInputImpl object was null. |

| **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** |
| --- |

### SQLInputImpl

public **SQLInputImpl**([Object](http://docs.google.com/java/lang/Object.html)[] attributes,  
 [Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),[Class](http://docs.google.com/java/lang/Class.html)<?>> map)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Creates an SQLInputImpl object initialized with the given array of attributes and the given type map. If any of the attributes is a UDT whose name is in an entry in the type map, the attribute will be mapped according to the corresponding SQLData implementation.

**Parameters:**attributes - an array of Object instances in which each element is an attribute of a UDT. The order of the attributes in the array is the same order in which the attributes were defined in the UDT definition.map - a java.util.Map object containing zero or more entries, with each entry consisting of 1) a String giving the fully qualified name of the UDT and 2) the Class object for the SQLData implementation that defines how the UDT is to be mapped **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the attributes or the map is a null value

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

### readString

public [String](http://docs.google.com/java/lang/String.html) **readString**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl object as a String in the Java programming language.

This method does not perform type-safe checking to determine if the returned type is the expected type; this responsibility is delegated to the UDT mapping as defined by a SQLData implementation.

**Specified by:**[readString](http://docs.google.com/java/sql/SQLInput.html#readString()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the next attribute in this SQLInputImpl object; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position or if there are no further values in the stream.

### readBoolean

public boolean **readBoolean**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl object as a boolean in the Java programming language.

This method does not perform type-safe checking to determine if the returned type is the expected type; this responsibility is delegated to the UDT mapping as defined by a SQLData implementation.

**Specified by:**[readBoolean](http://docs.google.com/java/sql/SQLInput.html#readBoolean()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the next attribute in this SQLInputImpl object; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position or if there are no further values in the stream.

### readByte

public byte **readByte**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl object as a byte in the Java programming language.

This method does not perform type-safe checking to determine if the returned type is the expected type; this responsibility is delegated to the UDT mapping as defined by a SQLData implementation.

**Specified by:**[readByte](http://docs.google.com/java/sql/SQLInput.html#readByte()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the next attribute in this SQLInputImpl object; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position or if there are no further values in the stream

### readShort

public short **readShort**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl object as a short in the Java programming language.

This method does not perform type-safe checking to determine if the returned type is the expected type; this responsibility is delegated to the UDT mapping as defined by a SQLData implementation.

**Specified by:**[readShort](http://docs.google.com/java/sql/SQLInput.html#readShort()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the next attribute in this SQLInputImpl object; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position or if there are no more values in the stream

### readInt

public int **readInt**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl object as an int in the Java programming language.

This method does not perform type-safe checking to determine if the returned type is the expected type; this responsibility is delegated to the UDT mapping as defined by a SQLData implementation.

**Specified by:**[readInt](http://docs.google.com/java/sql/SQLInput.html#readInt()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the next attribute in this SQLInputImpl object; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position or if there are no more values in the stream

### readLong

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

Retrieves the next attribute in this SQLInputImpl object as a long in the Java programming language.

This method does not perform type-safe checking to determine if the returned type is the expected type; this responsibility is delegated to the UDT mapping as defined by a SQLData implementation.

**Specified by:**[readLong](http://docs.google.com/java/sql/SQLInput.html#readLong()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the next attribute in this SQLInputImpl object; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position or if there are no more values in the stream

### readFloat

public float **readFloat**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl object as a float in the Java programming language.

This method does not perform type-safe checking to determine if the returned type is the expected type; this responsibility is delegated to the UDT mapping as defined by a SQLData implementation.

**Specified by:**[readFloat](http://docs.google.com/java/sql/SQLInput.html#readFloat()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the next attribute in this SQLInputImpl object; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position or if there are no more values in the stream

### readDouble

public double **readDouble**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl object as a double in the Java programming language.

This method does not perform type-safe checking to determine if the returned type is the expected type; this responsibility is delegated to the UDT mapping as defined by a SQLData implementation.

**Specified by:**[readDouble](http://docs.google.com/java/sql/SQLInput.html#readDouble()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the next attribute in this SQLInputImpl object; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position or if there are no more values in the stream

### readBigDecimal

public [BigDecimal](http://docs.google.com/java/math/BigDecimal.html) **readBigDecimal**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl object as a java.math.BigDecimal.

This method does not perform type-safe checking to determine if the returned type is the expected type; this responsibility is delegated to the UDT mapping as defined by a SQLData implementation.

**Specified by:**[readBigDecimal](http://docs.google.com/java/sql/SQLInput.html#readBigDecimal()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the next attribute in this SQLInputImpl object; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position or if there are no more values in the stream

### readBytes

public byte[] **readBytes**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl object as an array of bytes.

This method does not perform type-safe checking to determine if the returned type is the expected type; this responsibility is delegated to the UDT mapping as defined by a SQLData implementation.

**Specified by:**[readBytes](http://docs.google.com/java/sql/SQLInput.html#readBytes()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the next attribute in this SQLInputImpl object; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position or if there are no more values in the stream

### readDate

public [Date](http://docs.google.com/java/sql/Date.html) **readDate**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl as a java.sql.Date object.

This method does not perform type-safe checking to determine if the returned type is the expected type; this responsibility is delegated to the UDT mapping as defined by a SQLData implementation.

**Specified by:**[readDate](http://docs.google.com/java/sql/SQLInput.html#readDate()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the next attribute in this SQLInputImpl object; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position or if there are no more values in the stream

### readTime

public [Time](http://docs.google.com/java/sql/Time.html) **readTime**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl object as a java.sql.Time object.

This method does not perform type-safe checking to determine if the returned type is the expected type as this responsibility is delegated to the UDT mapping as implemented by a SQLData implementation.

**Specified by:**[readTime](http://docs.google.com/java/sql/SQLInput.html#readTime()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the attribute; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position; or if there are no further values in the stream.

### readTimestamp

public [Timestamp](http://docs.google.com/java/sql/Timestamp.html) **readTimestamp**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl object as a java.sql.Timestamp object.

**Specified by:**[readTimestamp](http://docs.google.com/java/sql/SQLInput.html#readTimestamp()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the attribute; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position; or if there are no further values in the stream.

### readCharacterStream

public [Reader](http://docs.google.com/java/io/Reader.html) **readCharacterStream**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the next attribute in this SQLInputImpl object as a stream of Unicode characters.

This method does not perform type-safe checking to determine if the returned type is the expected type as this responsibility is delegated to the UDT mapping as implemented by a SQLData implementation.

**Specified by:**[readCharacterStream](http://docs.google.com/java/sql/SQLInput.html#readCharacterStream()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the attribute; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position; or if there are no further values in the stream.

### readAsciiStream

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

Returns the next attribute in this SQLInputImpl object as a stream of ASCII characters.

This method does not perform type-safe checking to determine if the returned type is the expected type as this responsibility is delegated to the UDT mapping as implemented by a SQLData implementation.

**Specified by:**[readAsciiStream](http://docs.google.com/java/sql/SQLInput.html#readAsciiStream()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the attribute; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position; or if there are no further values in the stream.

### readBinaryStream

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

Returns the next attribute in this SQLInputImpl object as a stream of uninterpreted bytes.

This method does not perform type-safe checking to determine if the returned type is the expected type as this responsibility is delegated to the UDT mapping as implemented by a SQLData implementation.

**Specified by:**[readBinaryStream](http://docs.google.com/java/sql/SQLInput.html#readBinaryStream()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the attribute; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position; or if there are no further values in the stream.

### readObject

public [Object](http://docs.google.com/java/lang/Object.html) **readObject**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the value at the head of this SQLInputImpl object as an Object in the Java programming language. The actual type of the object returned is determined by the default mapping of SQL types to types in the Java programming language unless there is a custom mapping, in which case the type of the object returned is determined by this stream's type map.

The JDBC technology-enabled driver registers a type map with the stream before passing the stream to the application.

When the datum at the head of the stream is an SQL NULL, this method returns null. If the datum is an SQL structured or distinct type with a custom mapping, this method determines the SQL type of the datum at the head of the stream, constructs an object of the appropriate class, and calls the method SQLData.readSQL on that object. The readSQL method then calls the appropriate SQLInputImpl.readXXX methods to retrieve the attribute values from the stream.

**Specified by:**[readObject](http://docs.google.com/java/sql/SQLInput.html#readObject()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the value at the head of the stream as an Object in the Java programming language; null if the value is SQL NULL **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position; or if there are no further values in the stream.

### readRef

public [Ref](http://docs.google.com/java/sql/Ref.html) **readRef**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the value at the head of this SQLInputImpl object as a Ref object in the Java programming language.

**Specified by:**[readRef](http://docs.google.com/java/sql/SQLInput.html#readRef()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**a Ref object representing the SQL REF value at the head of the stream; if the value is SQL NULL return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position; or if there are no further values in the stream.

### readBlob

public [Blob](http://docs.google.com/java/sql/Blob.html) **readBlob**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the BLOB value at the head of this SQLInputImpl object as a Blob object in the Java programming language.

This method does not perform type-safe checking to determine if the returned type is the expected type as this responsibility is delegated to the UDT mapping as implemented by a SQLData implementation.

**Specified by:**[readBlob](http://docs.google.com/java/sql/SQLInput.html#readBlob()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**a Blob object representing the SQL BLOB value at the head of this stream; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position; or if there are no further values in the stream.

### readClob

public [Clob](http://docs.google.com/java/sql/Clob.html) **readClob**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the CLOB value at the head of this SQLInputImpl object as a Clob object in the Java programming language.

This method does not perform type-safe checking to determine if the returned type is the expected type as this responsibility is delegated to the UDT mapping as implemented by a SQLData implementation.

**Specified by:**[readClob](http://docs.google.com/java/sql/SQLInput.html#readClob()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**a Clob object representing the SQL CLOB value at the head of the stream; if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position; or if there are no further values in the stream.

### readArray

public [Array](http://docs.google.com/java/sql/Array.html) **readArray**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Reads an SQL ARRAY value from the stream and returns it as an Array object in the Java programming language.

This method does not perform type-safe checking to determine if the returned type is the expected type as this responsibility is delegated to the UDT mapping as implemented by a SQLData implementation.

**Specified by:**[readArray](http://docs.google.com/java/sql/SQLInput.html#readArray()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**an Array object representing the SQL ARRAY value at the head of the stream; \* if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position; or if there are no further values in the stream.

### wasNull

public boolean **wasNull**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Ascertains whether the last value read from this SQLInputImpl object was null.

**Specified by:**[wasNull](http://docs.google.com/java/sql/SQLInput.html#wasNull()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**true if the SQL value read most recently was null; otherwise, false; by default it will return false **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs determining the last value read was a null value or not;

### readURL

public [URL](http://docs.google.com/java/net/URL.html) **readURL**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Reads an SQL DATALINK value from the stream and returns it as an URL object in the Java programming language.

This method does not perform type-safe checking to determine if the returned type is the expected type as this responsibility is delegated to the UDT mapping as implemented by a SQLData implementation.

**Specified by:**[readURL](http://docs.google.com/java/sql/SQLInput.html#readURL()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**an URL object representing the SQL DATALINK value at the head of the stream; \* if the value is SQL NULL, return null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the read position is located at an invalid position; or if there are no further values in the stream.

### readNClob

public [NClob](http://docs.google.com/java/sql/NClob.html) **readNClob**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Reads an SQL NCLOB value from the stream and returns it as a Clob object in the Java programming language.

**Specified by:**[readNClob](http://docs.google.com/java/sql/SQLInput.html#readNClob()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**a NClob object representing data of the SQL NCLOB value at the head of the stream; null if the value read is SQL NULL **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs

### readNString

public [String](http://docs.google.com/java/lang/String.html) **readNString**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Reads the next attribute in the stream and returns it as a String in the Java programming language. It is intended for use when accessing NCHAR,NVARCHAR and LONGNVARCHAR columns.

**Specified by:**[readNString](http://docs.google.com/java/sql/SQLInput.html#readNString()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**the attribute; if the value is SQL NULL, returns null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs

### readSQLXML

public [SQLXML](http://docs.google.com/java/sql/SQLXML.html) **readSQLXML**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Reads an SQL XML value from the stream and returns it as a SQLXML object in the Java programming language.

**Specified by:**[readSQLXML](http://docs.google.com/java/sql/SQLInput.html#readSQLXML()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**a SQLXML object representing data of the SQL XML value at the head of the stream; null if the value read is SQL NULL **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs

### readRowId

public [RowId](http://docs.google.com/java/sql/RowId.html) **readRowId**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Reads an SQL ROWID value from the stream and returns it as a RowId object in the Java programming language.

**Specified by:**[readRowId](http://docs.google.com/java/sql/SQLInput.html#readRowId()) in interface [SQLInput](http://docs.google.com/java/sql/SQLInput.html) **Returns:**a RowId object representing data of the SQL ROWID value at the head of the stream; null if the value read is SQL NULL **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access 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/SQLInputImpl.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/SerialStruct.html)   [**NEXT CLASS**](http://docs.google.com/javax/sql/rowset/serial/SQLOutputImpl.html) | [**FRAMES**](http://docs.google.com/index.html?javax/sql/rowset/serial/SQLInputImpl.html)    [**NO FRAMES**](http://docs.google.com/SQLInputImpl.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

[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).