| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/RowSetMetaDataImpl.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/Predicate.html)   [**NEXT CLASS**](http://docs.google.com/javax/sql/rowset/RowSetWarning.html) | [**FRAMES**](http://docs.google.com/index.html?javax/sql/rowset/RowSetMetaDataImpl.html)    [**NO FRAMES**](http://docs.google.com/RowSetMetaDataImpl.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#4d34og8) | [METHOD](#17dp8vu) |

## **javax.sql.rowset**

Class RowSetMetaDataImpl

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

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html), [Wrapper](http://docs.google.com/java/sql/Wrapper.html), [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html)

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

Provides implementations for the methods that set and get metadata information about a RowSet object's columns. A RowSetMetaDataImpl object keeps track of the number of columns in the rowset and maintains an internal array of column attributes for each column.

A RowSet object creates a RowSetMetaDataImpl object internally in order to set and retrieve information about its columns.

NOTE: All metadata in a RowSetMetaDataImpl object should be considered as unavailable until the RowSet object that it describes is populated. Therefore, any RowSetMetaDataImpl method that retrieves information is defined as having unspecified behavior when it is called before the RowSet object contains data.

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

| **Field Summary** | |
| --- | --- |

| **Fields inherited from interface java.sql.**[**ResultSetMetaData**](http://docs.google.com/java/sql/ResultSetMetaData.html) |
| --- |
| [columnNoNulls](http://docs.google.com/java/sql/ResultSetMetaData.html#columnNoNulls), [columnNullable](http://docs.google.com/java/sql/ResultSetMetaData.html#columnNullable), [columnNullableUnknown](http://docs.google.com/java/sql/ResultSetMetaData.html#columnNullableUnknown) |

| **Constructor Summary** | |
| --- | --- |
| [**RowSetMetaDataImpl**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#RowSetMetaDataImpl())() |

| **Method Summary** | |
| --- | --- |
| [String](http://docs.google.com/java/lang/String.html) | [**getCatalogName**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#getCatalogName(int))(int columnIndex)            Retrieves the catalog name of the table from which the value in the designated column was derived. |
| [String](http://docs.google.com/java/lang/String.html) | [**getColumnClassName**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#getColumnClassName(int))(int columnIndex)            Retrieves the fully-qualified name of the class in the Java programming language to which a value in the designated column will be mapped. |
| int | [**getColumnCount**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#getColumnCount())()            Retrieves the number of columns in the RowSet object for which this RowSetMetaDataImpl object was created. |
| int | [**getColumnDisplaySize**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#getColumnDisplaySize(int))(int columnIndex)            Retrieves the normal maximum width in chars of the designated column. |
| [String](http://docs.google.com/java/lang/String.html) | [**getColumnLabel**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#getColumnLabel(int))(int columnIndex)            Retrieves the the suggested column title for the designated column for use in printouts and displays. |
| [String](http://docs.google.com/java/lang/String.html) | [**getColumnName**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#getColumnName(int))(int columnIndex)            Retrieves the name of the designated column. |
| int | [**getColumnType**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#getColumnType(int))(int columnIndex)            Retrieves the type code (one of the java.sql.Types constants) for the SQL type of the value stored in the designated column. |
| [String](http://docs.google.com/java/lang/String.html) | [**getColumnTypeName**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#getColumnTypeName(int))(int columnIndex)            Retrieves the DBMS-specific type name for values stored in the designated column. |
| int | [**getPrecision**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#getPrecision(int))(int columnIndex)            Retrieves the total number of digits for values stored in the designated column. |
| int | [**getScale**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#getScale(int))(int columnIndex)            Retrieves the number of digits to the right of the decimal point for values stored in the designated column. |
| [String](http://docs.google.com/java/lang/String.html) | [**getSchemaName**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#getSchemaName(int))(int columnIndex)            Retrieves the schema name of the table from which the value in the designated column was derived. |
| [String](http://docs.google.com/java/lang/String.html) | [**getTableName**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#getTableName(int))(int columnIndex)            Retrieves the name of the table from which the value in the designated column was derived. |
| boolean | [**isAutoIncrement**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#isAutoIncrement(int))(int columnIndex)            Retrieves whether a value stored in the designated column is automatically numbered, and thus readonly. |
| boolean | [**isCaseSensitive**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#isCaseSensitive(int))(int columnIndex)            Indicates whether the case of the designated column's name matters. |
| boolean | [**isCurrency**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#isCurrency(int))(int columnIndex)            Indicates whether a value stored in the designated column is a cash value. |
| boolean | [**isDefinitelyWritable**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#isDefinitelyWritable(int))(int columnIndex)            Indicates whether a write operation on the designated column will definitely succeed. |
| int | [**isNullable**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#isNullable(int))(int columnIndex)            Retrieves a constant indicating whether it is possible to store a NULL value in the designated column. |
| boolean | [**isReadOnly**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#isReadOnly(int))(int columnIndex)            Indicates whether the designated column is definitely not writable, thus readonly. |
| boolean | [**isSearchable**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#isSearchable(int))(int columnIndex)            Indicates whether a value stored in the designated column can be used in a WHERE clause. |
| boolean | [**isSigned**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#isSigned(int))(int columnIndex)            Indicates whether a value stored in the designated column is a signed number. |
| boolean | [**isWrapperFor**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#isWrapperFor(java.lang.Class))([Class](http://docs.google.com/java/lang/Class.html)<?> interfaces)            Returns true if this either implements the interface argument or is directly or indirectly a wrapper for an object that does. |
| boolean | [**isWritable**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#isWritable(int))(int columnIndex)            Indicates whether it is possible for a write operation on the designated column to succeed. |
| void | [**setAutoIncrement**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setAutoIncrement(int,%20boolean))(int columnIndex, boolean property)            Sets whether the designated column is automatically numbered, thus read-only, to the given boolean value. |
| void | [**setCaseSensitive**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setCaseSensitive(int,%20boolean))(int columnIndex, boolean property)            Sets whether the name of the designated column is case sensitive to the given boolean. |
| void | [**setCatalogName**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setCatalogName(int,%20java.lang.String))(int columnIndex, [String](http://docs.google.com/java/lang/String.html) catalogName)            Sets the catalog name of the table from which the designated column was derived to *catalogName*. |
| void | [**setColumnCount**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setColumnCount(int))(int columnCount)            Sets to the given number the number of columns in the RowSet object for which this RowSetMetaDataImpl object was created. |
| void | [**setColumnDisplaySize**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setColumnDisplaySize(int,%20int))(int columnIndex, int size)            Sets the normal maximum number of chars in the designated column to the given number. |
| void | [**setColumnLabel**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setColumnLabel(int,%20java.lang.String))(int columnIndex, [String](http://docs.google.com/java/lang/String.html) label)            Sets the suggested column label for use in printouts and displays, if any, to *label*. |
| void | [**setColumnName**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setColumnName(int,%20java.lang.String))(int columnIndex, [String](http://docs.google.com/java/lang/String.html) columnName)            Sets the column name of the designated column to the given name. |
| void | [**setColumnType**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setColumnType(int,%20int))(int columnIndex, int SQLType)            Sets the SQL type code for values stored in the designated column to the given type code from the class java.sql.Types. |
| void | [**setColumnTypeName**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setColumnTypeName(int,%20java.lang.String))(int columnIndex, [String](http://docs.google.com/java/lang/String.html) typeName)            Sets the type name used by the data source for values stored in the designated column to the given type name. |
| void | [**setCurrency**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setCurrency(int,%20boolean))(int columnIndex, boolean property)            Sets whether a value stored in the designated column is a cash value to the given boolean. |
| void | [**setNullable**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setNullable(int,%20int))(int columnIndex, int property)            Sets whether a value stored in the designated column can be set to NULL to the given constant from the interface ResultSetMetaData. |
| void | [**setPrecision**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setPrecision(int,%20int))(int columnIndex, int precision)            Sets the total number of decimal digits in a value stored in the designated column to the given number. |
| void | [**setScale**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setScale(int,%20int))(int columnIndex, int scale)            Sets the number of digits to the right of the decimal point in a value stored in the designated column to the given number. |
| void | [**setSchemaName**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setSchemaName(int,%20java.lang.String))(int columnIndex, [String](http://docs.google.com/java/lang/String.html) schemaName)            Sets the designated column's table's schema name, if any, to *schemaName*. |
| void | [**setSearchable**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setSearchable(int,%20boolean))(int columnIndex, boolean property)            Sets whether a value stored in the designated column can be used in a WHERE clause to the given boolean value. |
| void | [**setSigned**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setSigned(int,%20boolean))(int columnIndex, boolean property)            Sets whether a value stored in the designated column is a signed number to the given boolean. |
| void | [**setTableName**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#setTableName(int,%20java.lang.String))(int columnIndex, [String](http://docs.google.com/java/lang/String.html) tableName)            Sets the name of the table from which the designated column was derived to the given table name. |
| | <T> T | | --- | | [**unwrap**](http://docs.google.com/javax/sql/rowset/RowSetMetaDataImpl.html#unwrap(java.lang.Class))([Class](http://docs.google.com/java/lang/Class.html)<T> iface)            Returns an object that implements the given interface to allow access to non-standard methods, or standard methods not exposed by the proxy. |

| **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** |
| --- |

### RowSetMetaDataImpl

public **RowSetMetaDataImpl**()

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

### setColumnCount

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

Sets to the given number the number of columns in the RowSet object for which this RowSetMetaDataImpl object was created.

**Specified by:**[setColumnCount](http://docs.google.com/javax/sql/RowSetMetaData.html#setColumnCount(int)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnCount - an int giving the number of columns in the RowSet object **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if the given number is equal to or less than zero

### setAutoIncrement

public void **setAutoIncrement**(int columnIndex,  
 boolean property)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets whether the designated column is automatically numbered, thus read-only, to the given boolean value.

**Specified by:**[setAutoIncrement](http://docs.google.com/javax/sql/RowSetMetaData.html#setAutoIncrement(int,%20boolean)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns in the rowset, inclusiveproperty - true if the given column is automatically incremented; false otherwise **Throws:** SQLException - if a database access error occurs or the given index is out of bounds [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs

### setCaseSensitive

public void **setCaseSensitive**(int columnIndex,  
 boolean property)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets whether the name of the designated column is case sensitive to the given boolean.

**Specified by:**[setCaseSensitive](http://docs.google.com/javax/sql/RowSetMetaData.html#setCaseSensitive(int,%20boolean)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns in the rowset, inclusiveproperty - true to indicate that the column name is case sensitive; false otherwise **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### setSearchable

public void **setSearchable**(int columnIndex,  
 boolean property)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets whether a value stored in the designated column can be used in a WHERE clause to the given boolean value.

**Specified by:**[setSearchable](http://docs.google.com/javax/sql/RowSetMetaData.html#setSearchable(int,%20boolean)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns in the rowset, inclusiveproperty - true to indicate that a column value can be used in a WHERE clause; false otherwise **Throws:** SQLException - if a database access error occurs or the given column number is out of bounds [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs

### setCurrency

public void **setCurrency**(int columnIndex,  
 boolean property)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets whether a value stored in the designated column is a cash value to the given boolean.

**Specified by:**[setCurrency](http://docs.google.com/javax/sql/RowSetMetaData.html#setCurrency(int,%20boolean)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive between 1 and the number of columns, inclusiveproperty - true if the value is a cash value; false otherwise. **Throws:** SQLException - if a database access error occurs or the given column number is out of bounds [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs

### setNullable

public void **setNullable**(int columnIndex,  
 int property)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets whether a value stored in the designated column can be set to NULL to the given constant from the interface ResultSetMetaData.

**Specified by:**[setNullable](http://docs.google.com/javax/sql/RowSetMetaData.html#setNullable(int,%20int)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusiveproperty - one of the following ResultSetMetaData constants: columnNoNulls, columnNullable, or columnNullableUnknown **Throws:** SQLException - if a database access error occurs, the given column number is out of bounds, or the value supplied for the *property* parameter is not one of the following constants: ResultSetMetaData.columnNoNulls, ResultSetMetaData.columnNullable, or ResultSetMetaData.columnNullableUnknown [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs

### setSigned

public void **setSigned**(int columnIndex,  
 boolean property)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets whether a value stored in the designated column is a signed number to the given boolean.

**Specified by:**[setSigned](http://docs.google.com/javax/sql/RowSetMetaData.html#setSigned(int,%20boolean)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusiveproperty - true to indicate that a column value is a signed number; false to indicate that it is not **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### setColumnDisplaySize

public void **setColumnDisplaySize**(int columnIndex,  
 int size)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets the normal maximum number of chars in the designated column to the given number.

**Specified by:**[setColumnDisplaySize](http://docs.google.com/javax/sql/RowSetMetaData.html#setColumnDisplaySize(int,%20int)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusivesize - the maximum size of the column in chars; must be 0 or more **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs, the given column number is out of bounds, or *size* is less than 0

### setColumnLabel

public void **setColumnLabel**(int columnIndex,  
 [String](http://docs.google.com/java/lang/String.html) label)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets the suggested column label for use in printouts and displays, if any, to *label*. If *label* is null, the column label is set to an empty string ("").

**Specified by:**[setColumnLabel](http://docs.google.com/javax/sql/RowSetMetaData.html#setColumnLabel(int,%20java.lang.String)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusivelabel - the column label to be used in printouts and displays; if the column label is null, an empty String is set **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column index is out of bounds

### setColumnName

public void **setColumnName**(int columnIndex,  
 [String](http://docs.google.com/java/lang/String.html) columnName)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets the column name of the designated column to the given name.

**Specified by:**[setColumnName](http://docs.google.com/javax/sql/RowSetMetaData.html#setColumnName(int,%20java.lang.String)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusivecolumnName - a String object indicating the column name; if the given name is null, an empty String is set **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column index is out of bounds

### setSchemaName

public void **setSchemaName**(int columnIndex,  
 [String](http://docs.google.com/java/lang/String.html) schemaName)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets the designated column's table's schema name, if any, to *schemaName*. If *schemaName* is null, the schema name is set to an empty string ("").

**Specified by:**[setSchemaName](http://docs.google.com/javax/sql/RowSetMetaData.html#setSchemaName(int,%20java.lang.String)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusiveschemaName - the schema name for the table from which a value in the designated column was derived; may be an empty String or null **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### setPrecision

public void **setPrecision**(int columnIndex,  
 int precision)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets the total number of decimal digits in a value stored in the designated column to the given number.

**Specified by:**[setPrecision](http://docs.google.com/javax/sql/RowSetMetaData.html#setPrecision(int,%20int)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusiveprecision - the total number of decimal digits; must be 0 or more **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs, *columnIndex* is out of bounds, or *precision* is less than 0

### setScale

public void **setScale**(int columnIndex,  
 int scale)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets the number of digits to the right of the decimal point in a value stored in the designated column to the given number.

**Specified by:**[setScale](http://docs.google.com/javax/sql/RowSetMetaData.html#setScale(int,%20int)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusivescale - the number of digits to the right of the decimal point; must be zero or greater **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs, *columnIndex* is out of bounds, or *scale* is less than 0

### setTableName

public void **setTableName**(int columnIndex,  
 [String](http://docs.google.com/java/lang/String.html) tableName)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets the name of the table from which the designated column was derived to the given table name.

**Specified by:**[setTableName](http://docs.google.com/javax/sql/RowSetMetaData.html#setTableName(int,%20java.lang.String)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusivetableName - the column's table name; may be null or an empty string **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### setCatalogName

public void **setCatalogName**(int columnIndex,  
 [String](http://docs.google.com/java/lang/String.html) catalogName)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets the catalog name of the table from which the designated column was derived to *catalogName*. If *catalogName* is null, the catalog name is set to an empty string.

**Specified by:**[setCatalogName](http://docs.google.com/javax/sql/RowSetMetaData.html#setCatalogName(int,%20java.lang.String)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusivecatalogName - the column's table's catalog name; if the catalogName is null, an empty String is set **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### setColumnType

public void **setColumnType**(int columnIndex,  
 int SQLType)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets the SQL type code for values stored in the designated column to the given type code from the class java.sql.Types.

**Specified by:**[setColumnType](http://docs.google.com/javax/sql/RowSetMetaData.html#setColumnType(int,%20int)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusiveSQLType - the designated column's SQL type, which must be one of the constants in the class java.sql.Types **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs, the given column number is out of bounds, or the column type specified is not one of the constants in java.sql.Types**See Also:**[Types](http://docs.google.com/java/sql/Types.html)

### setColumnTypeName

public void **setColumnTypeName**(int columnIndex,  
 [String](http://docs.google.com/java/lang/String.html) typeName)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Sets the type name used by the data source for values stored in the designated column to the given type name.

**Specified by:**[setColumnTypeName](http://docs.google.com/javax/sql/RowSetMetaData.html#setColumnTypeName(int,%20java.lang.String)) in interface [RowSetMetaData](http://docs.google.com/javax/sql/RowSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusivetypeName - the data source-specific type name; if *typeName* is null, an empty String is set **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### getColumnCount

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

Retrieves the number of columns in the RowSet object for which this RowSetMetaDataImpl object was created.

**Specified by:**[getColumnCount](http://docs.google.com/java/sql/ResultSetMetaData.html#getColumnCount()) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Returns:**the number of columns **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs determining the column count

### isAutoIncrement

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

Retrieves whether a value stored in the designated column is automatically numbered, and thus readonly.

**Specified by:**[isAutoIncrement](http://docs.google.com/java/sql/ResultSetMetaData.html#isAutoIncrement(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**true if the column is automatically numbered; false otherwise **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### isCaseSensitive

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

Indicates whether the case of the designated column's name matters.

**Specified by:**[isCaseSensitive](http://docs.google.com/java/sql/ResultSetMetaData.html#isCaseSensitive(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**true if the column name is case sensitive; false otherwise **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### isSearchable

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

Indicates whether a value stored in the designated column can be used in a WHERE clause.

**Specified by:**[isSearchable](http://docs.google.com/java/sql/ResultSetMetaData.html#isSearchable(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**true if a value in the designated column can be used in a WHERE clause; false otherwise **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### isCurrency

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

Indicates whether a value stored in the designated column is a cash value.

**Specified by:**[isCurrency](http://docs.google.com/java/sql/ResultSetMetaData.html#isCurrency(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**true if a value in the designated column is a cash value; false otherwise **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### isNullable

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

Retrieves a constant indicating whether it is possible to store a NULL value in the designated column.

**Specified by:**[isNullable](http://docs.google.com/java/sql/ResultSetMetaData.html#isNullable(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**a constant from the ResultSetMetaData interface; either columnNoNulls, columnNullable, or columnNullableUnknown **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### isSigned

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

Indicates whether a value stored in the designated column is a signed number.

**Specified by:**[isSigned](http://docs.google.com/java/sql/ResultSetMetaData.html#isSigned(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**true if if a value in the designated column is a signed number; false otherwise **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### getColumnDisplaySize

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

Retrieves the normal maximum width in chars of the designated column.

**Specified by:**[getColumnDisplaySize](http://docs.google.com/java/sql/ResultSetMetaData.html#getColumnDisplaySize(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**the maximum number of chars that can be displayed in the designated column **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### getColumnLabel

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

Retrieves the the suggested column title for the designated column for use in printouts and displays.

**Specified by:**[getColumnLabel](http://docs.google.com/java/sql/ResultSetMetaData.html#getColumnLabel(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**the suggested column name to use in printouts and displays **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### getColumnName

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

Retrieves the name of the designated column.

**Specified by:**[getColumnName](http://docs.google.com/java/sql/ResultSetMetaData.html#getColumnName(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**the column name of the designated column **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### getSchemaName

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

Retrieves the schema name of the table from which the value in the designated column was derived.

**Specified by:**[getSchemaName](http://docs.google.com/java/sql/ResultSetMetaData.html#getSchemaName(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**the schema name or an empty String if no schema name is available **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### getPrecision

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

Retrieves the total number of digits for values stored in the designated column.

**Specified by:**[getPrecision](http://docs.google.com/java/sql/ResultSetMetaData.html#getPrecision(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**the precision for values stored in the designated column **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### getScale

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

Retrieves the number of digits to the right of the decimal point for values stored in the designated column.

**Specified by:**[getScale](http://docs.google.com/java/sql/ResultSetMetaData.html#getScale(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**the scale for values stored in the designated column **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### getTableName

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

Retrieves the name of the table from which the value in the designated column was derived.

**Specified by:**[getTableName](http://docs.google.com/java/sql/ResultSetMetaData.html#getTableName(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**the table name or an empty String if no table name is available **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### getCatalogName

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

Retrieves the catalog name of the table from which the value in the designated column was derived.

**Specified by:**[getCatalogName](http://docs.google.com/java/sql/ResultSetMetaData.html#getCatalogName(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**the catalog name of the column's table or an empty String if no catalog name is available **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### getColumnType

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

Retrieves the type code (one of the java.sql.Types constants) for the SQL type of the value stored in the designated column.

**Specified by:**[getColumnType](http://docs.google.com/java/sql/ResultSetMetaData.html#getColumnType(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**an int representing the SQL type of values stored in the designated column **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds**See Also:**[Types](http://docs.google.com/java/sql/Types.html)

### getColumnTypeName

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

Retrieves the DBMS-specific type name for values stored in the designated column.

**Specified by:**[getColumnTypeName](http://docs.google.com/java/sql/ResultSetMetaData.html#getColumnTypeName(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**the type name used by the data source **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### isReadOnly

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

Indicates whether the designated column is definitely not writable, thus readonly.

**Specified by:**[isReadOnly](http://docs.google.com/java/sql/ResultSetMetaData.html#isReadOnly(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**true if this RowSet object is read-Only and thus not updatable; false otherwise **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### isWritable

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

Indicates whether it is possible for a write operation on the designated column to succeed. A return value of true means that a write operation may or may not succeed.

**Specified by:**[isWritable](http://docs.google.com/java/sql/ResultSetMetaData.html#isWritable(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**true if a write operation on the designated column may will succeed; false otherwise **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### isDefinitelyWritable

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

Indicates whether a write operation on the designated column will definitely succeed.

**Specified by:**[isDefinitelyWritable](http://docs.google.com/java/sql/ResultSetMetaData.html#isDefinitelyWritable(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**true if a write operation on the designated column will definitely succeed; false otherwise **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### getColumnClassName

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

Retrieves the fully-qualified name of the class in the Java programming language to which a value in the designated column will be mapped. For example, if the value is an int, the class name returned by this method will be java.lang.Integer.

If the value in the designated column has a custom mapping, this method returns the name of the class that implements SQLData. When the method ResultSet.getObject is called to retrieve a value from the designated column, it will create an instance of this class or one of its subclasses.

**Specified by:**[getColumnClassName](http://docs.google.com/java/sql/ResultSetMetaData.html#getColumnClassName(int)) in interface [ResultSetMetaData](http://docs.google.com/java/sql/ResultSetMetaData.html) **Parameters:**columnIndex - the first column is 1, the second is 2, and so on; must be between 1 and the number of columns, inclusive **Returns:**the fully-qualified name of the class in the Java programming language that would be used by the method RowSet.getObject to retrieve the value in the specified column. This is the class name used for custom mapping when there is a custom mapping. **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs or the given column number is out of bounds

### unwrap

public <T> T **unwrap**([Class](http://docs.google.com/java/lang/Class.html)<T> iface)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Returns an object that implements the given interface to allow access to non-standard methods, or standard methods not exposed by the proxy. The result may be either the object found to implement the interface or a proxy for that object. If the receiver implements the interface then that is the object. If the receiver is a wrapper and the wrapped object implements the interface then that is the object. Otherwise the object is the result of calling unwrap recursively on the wrapped object. If the receiver is not a wrapper and does not implement the interface, then an SQLException is thrown.

**Specified by:**[unwrap](http://docs.google.com/java/sql/Wrapper.html#unwrap(java.lang.Class)) in interface [Wrapper](http://docs.google.com/java/sql/Wrapper.html) **Parameters:**iface - A Class defining an interface that the result must implement. **Returns:**an object that implements the interface. May be a proxy for the actual implementing object. **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - If no object found that implements the interface**Since:** 1.6

### isWrapperFor

public boolean **isWrapperFor**([Class](http://docs.google.com/java/lang/Class.html)<?> interfaces)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Returns true if this either implements the interface argument or is directly or indirectly a wrapper for an object that does. Returns false otherwise. If this implements the interface then return true, else if this is a wrapper then return the result of recursively calling isWrapperFor on the wrapped object. If this does not implement the interface and is not a wrapper, return false. This method should be implemented as a low-cost operation compared to unwrap so that callers can use this method to avoid expensive unwrap calls that may fail. If this method returns true then calling unwrap with the same argument should succeed.

**Specified by:**[isWrapperFor](http://docs.google.com/java/sql/Wrapper.html#isWrapperFor(java.lang.Class)) in interface [Wrapper](http://docs.google.com/java/sql/Wrapper.html) **Parameters:**interfaces - a Class defining an interface. **Returns:**true if this implements the interface or directly or indirectly wraps an object that does. **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs while determining whether this is a wrapper for an object with the given interface.**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/RowSetMetaDataImpl.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/Predicate.html)   [**NEXT CLASS**](http://docs.google.com/javax/sql/rowset/RowSetWarning.html) | [**FRAMES**](http://docs.google.com/index.html?javax/sql/rowset/RowSetMetaDataImpl.html)    [**NO FRAMES**](http://docs.google.com/RowSetMetaDataImpl.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#4d34og8) | [METHOD](#17dp8vu) |

[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).