

# JSColumn

## Constants Summery

- Number [#DATETIME](#)  
Constant used when setting or getting the type of columns.
- Number [#INTEGER](#)  
Constant used when setting or getting the type of columns.
- Number [#MEDIA](#)  
Constant used when setting or getting the type of columns.
- Number [#NUMBER](#)  
Constant used when setting or getting the type of columns.
- Number [#TEXT](#)  
Constant used when setting or getting the type of columns.

## Method Summery

- Boolean [#getAllowNull\(\)](#)  
Get the allow-null flag of the column.
- String [#getDataProviderID\(\)](#)  
Get the data provider id (name) for this column.
- String [#getDescription\(\)](#)  
Get the description property of the column.
- String [#getForeignType\(\)](#)  
Get the foreign type of the column.
- Number [#getLength\(\)](#)  
Get the length of the column as reported by the JDBC driver.
- String [#getQualifiedName\(\)](#)  
Get the qualified name (including table name) of the column as known by the database.
- String [#getQuotedSQLName\(\)](#)  
Returns a quoted version of the column name, if necessary, as defined by the actual database used.
- String [#getSQLName\(\)](#)  
Get the name of the column as known by the database.
- Number [#getScale\(\)](#)  
Get the scale of the column as reported by the JDBC driver.
- String [#getTitle\(\)](#)  
Get the title property of the column.
- Number [#getType\(\)](#)  
Get the JDBC type of the column.
- String [#getTypeAsString\(\)](#)  
Get the name JDBC type of the column.
- Boolean [#isRowIdentifier\(\)](#)  
Is this column one of the row identifiers for its table.
- Boolean [#isUUID\(\)](#)  
Is this column marked as UUID column.

## Constants Details

## DATETIME

Constant used when setting or getting the type of columns.

### Returns

[Number](#)

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customerid')
switch (column.getType())
{
    case JSColumn.TEXT:
        // handle text column
        break;

    case JSColumn.NUMBER:
    case JSColumn.INTEGER:
        // handle numerical column
        break;
}
```

## INTEGER

Constant used when setting or getting the type of columns.

### Returns

[Number](#)

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customerid')
switch (column.getType())
{
    case JSColumn.TEXT:
        // handle text column
        break;

    case JSColumn.NUMBER:
    case JSColumn.INTEGER:
        // handle numerical column
        break;
}
```

## MEDIA

Constant used when setting or getting the type of columns.

### Returns

[Number](#)

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customerid')
switch (column.getType())
{
    case JSColumn.TEXT:
        // handle text column
        break;

    case JSColumn.NUMBER:
    case JSColumn.INTEGER:
        // handle numerical column
        break;
}
```

## NUMBER

Constant used when setting or getting the type of columns.

### Returns

Number

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customerid')
switch (column.getType())
{
    case JSColumn.TEXT:
        // handle text column
        break;

    case JSColumn.NUMBER:
    case JSColumn.INTEGER:
        // handle numerical column
        break;
}
```

## TEXT

Constant used when setting or getting the type of columns.

### Returns

Number

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customerid')
switch (column.getType())
{
    case JSColumn.TEXT:
        // handle text column
        break;

    case JSColumn.NUMBER:
    case JSColumn.INTEGER:
        // handle numerical column
        break;
}
```

## Method Details

getAllowNull

Boolean **getAllowNull()**

Get the allow-null flag of the column.

### Returns

Boolean – boolean allow-null flag.

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customerid')
if (!column.getAllowNull())
{
    // column cannot be null
}
```

getDataProviderID

String **getDataProviderID()**

Get the data provider id (name) for this column.

### Returns

String – String dataprovider id.

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customerid')
var dataProviderId = column.getDataProviderID()
```

getDescription

[String](#) **getDescription()**

Get the description property of the column.

#### Returns

[String](#) – String column description.

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customername')
var desc = column.getDescription()
```

getForeignType

[String](#) **getForeignType()**

Get the foreign type of the column.

The foreign type can be defined design time as a foreign key reference to another table.

#### Returns

[String](#) – String foreign type.

#### Also see

[databaseManager.mergeRecords](#)

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customerid')
var foreignType = column.getForeignType()
if (foreignType != null)
{
    var fkTable = databaseManager.getTable('example_data', foreignType)
}
```

getLength

[Number](#) **getLength()**

Get the length of the column as reported by the JDBC driver.

#### Returns

[Number](#) – int column length.

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customername')
if (column.getLength() < 10)
{
    // handle short column
}
```

getQualifiedName

[String](#) **getQualifiedName()**

Get the qualified name (including table name) of the column as known by the database.

The name is quoted, if necessary, as defined by the actual database used.

#### Returns

[String](#) – String qualified column name.

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customerid')
var qualifiedSqlName = column.getQualifiedName()
```

getQuotedSQLName

**String** getQuotedSQLName()

Returns a quoted version of the column name, if necessary, as defined by the actual database used.

**Returns**

**String** – column name, quoted if needed.

**Sample**

```
//use with the raw SQL plugin:
//if the table name contains characters that are illegal in sql, the table name will be quoted
var jsTable = databaseManager.getTable('udm', 'campaigns')
var quotedTableName = jsTable.getQuotedSQLName()
var jsColumn = jsTable.getColumn('active')
var quotedColumnName = jsColumn.getQuotedSQLName()
plugins.rawSQL.executeSQL('udm', quotedTableName, 'select * from ' + quotedTableName + ' where ' +
quotedColumnName + ' = ?', [1])
```

getSQLName

**String** getSQLName()

Get the name of the column as known by the database.

**Returns**

**String** – String sql name

**Sample**

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customerid')
var sqlName = column.getSQLName()
```

getScale

**Number** getScale()

Get the scale of the column as reported by the JDBC driver.

**Returns**

**Number** – int column scale.

**Sample**

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customername')
var scale = column.getScale()
```

getTitle

**String** getTitle()

Get the title property of the column.

**Returns**

**String** – String column title.

**Sample**

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customername')
var title = column.getTitle()
```

getType

**Number** getType()

Get the JDBC type of the column.

The type reported by the JDBC driver will be mapped to one of:

- JSColumn.DATETIME
- JSColumn.TEXT
- JSColumn.NUMBER
- JSColumn.INTEGER
- JSColumn.MEDIA

**Returns**

**Number** – int sql type.

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customerid')
switch (column.getType())
{
    case JSColumn.TEXT:
        // handle text column
        break;

    case JSColumn.NUMBER:
    case JSColumn.INTEGER:
        // handle numerical column
        break;
}
```

### getTypeAsString

#### **String** [getTypeAsString\(\)](#)

Get the name JDBC type of the column.

The same mapping as defined in [JSColumn.getType\(\)](#) is applied.

#### **Returns**

[String](#) – String sql name.

#### **Also see**

[JSColumn.getType](#)

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('customerid')
var typeName = column.getTypeAsString()
```

### isRowIdentifier

#### **Boolean** [isRowIdentifier\(\)](#)

Is this column one of the row identifiers for its table.

#### **Returns**

[Boolean](#) – boolean true if is row identifier else false.

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('orderid')
if (column.isRowIdentifier())
{
    // handle pk column
}
```

### isUUID

#### **Boolean** [isUUID\(\)](#)

Is this column marked as UUID column.

#### **Returns**

[Boolean](#) – boolean true if is marked as UUID columns else false.

### Sample

```
var table = databaseManager.getTable('example_data', 'orders')
var column = table.getColumn('orderid')
if (column.isUUID())
{
    // handle UUID column
}
```