

# JSRecord

## Property Summary

[ServoyException](#) [#exception](#)  
Returns last occurred exception on this record (or null).

[JSFoundSet](#) [#foundset](#)  
Returns parent foundset of the record.

## Method Summary

[JSDataSet](#) [#getChangedData\(\)](#)  
Returns a JSDataSet with outstanding (not saved) changed data of this record.

[String](#) [#getDataSource\(\)](#)  
Returns the records datasource string.

[Object\[\]](#) [#getPKs\(\)](#)  
Returns an array with the primary key values of the record.

[Boolean](#) [#isEditing\(\)](#)  
Returns true or false if the record is being edited or not.

[void](#) [#revertChanges\(\)](#)  
Reverts the in memory outstanding (not saved) changes of the record.

## Property Details

[exception](#)  
Returns last occurred exception on this record (or null).

### Returns

[ServoyException](#) – The occurred exception.

### Sample

```
var exception = record.exception;
```

[foundset](#)  
Returns parent foundset of the record.

### Returns

[JSFoundSet](#) – The parent foundset of the record.

### Sample

```
var parent = record.foundset;
```

## Method Details

[getChangedData](#)

[JSDataSet](#) [getChangedData\(\)](#)

Returns a JSDataSet with outstanding (not saved) changed data of this record.  
column1 is the column name, column2 is the old data and column3 is the new data.

NOTE: To return an array of records with outstanding changed data, see the function [databaseManager.getEditedRecords\(\)](#).

### Returns

[JSDataSet](#) – a JSDataSet with the changed data of this record.

### Sample

```
/** @type {JSDataSet} */
var dataset = record.getChangedData()
for( var i = 1 ; i <= dataset.getMaxRowIndex() ; i++ )
{
    application.output(dataset.getValue(i,1) + ' ' + dataset.getValue(i,2) + ' ' + dataset.getValue(i,3));
}
```

getDataSource

**String** **getDataSource()**

Returns the records datasource string.

**Returns**

**String** – The datasource string of this record.

**Sample**

```
var ds = record.getDataSource();
```

getPKs

**Object[]** **getPKs()**

Returns an array with the primary key values of the record.

**Returns**

**Object[]** – an Array with the pk values.

**Sample**

```
var pks = foundset.getSelectedRecord().getPKs() // also foundset.getRecord can be used
```

isEditing

**Boolean** **isEditing()**

Returns true or false if the record is being edited or not.

**Returns**

**Boolean** – a boolean when in edit.

**Sample**

```
var isEditing = foundset.getSelectedRecord().isEditing() // also foundset.getRecord can be used
```

revertChanges

**void** **revertChanges()**

Reverts the in memory outstanding (not saved) changes of the record.

**Returns**

**void**

**Sample**

```
var record= forms.customer.foundset.getSelectedRecord();
record.revertChanges();
```