JSRecord

Feb 17, 2024 14:35

Supported Clients

SmartClient WebClient NGClient MobileClient

Property Summary

Exception	exception	Returns last occurred exception on this record (or null).
JSFoundSet	foundset	Returns parent foundset of the record.
JSRecordMarkers	recordMarkers	Returns the validation object if there where validation failures for this record Can be set to null again if you checked the problems, will also be set to null when a save was successful.

Methods Summary

JSRecordMarkers	createMarkers()	Creates and returns a new validation object for this record, which allows for markers to be used outside the validation flow.
JSDataSet	getChangedData()	Returns a JSDataSet with outstanding (not saved) changed data of this record.
String	getDataSource()	Returns the records datasource string.
Array	getPKs()	Returns an array with the primary key values of the record.
Boolean	hasChangedData()	Returns true if the current record has outstanding/changed data.
Boolean	isEditing()	Returns true or false if the record is being edited or not.
Boolean	isNew()	Returns true if the current record is a new record or false otherwise.
Boolean	isRelatedFoundSetLoaded(relationName)	Returns true or false if the related foundset is already loaded.
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

Exception The occurred exception.

Supported Clients

SmartClient,WebClient,NGClient

Sample

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

foundset

```
Returns parent foundset of the record.
```

Returns

JSFoundSet The parent foundset of the record.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

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

recordMarkers

```
Returns the validation object if there where validation failures for this record
Can be set to null again if you checked the problems, will also be set to null when a save was successful.
```

Returns

JSRecordMarkers The last validation object if the record was not validated.

Supported Clients

SmartClient,WebClient,NGClient

Sample

```
var recordMarkers = record.recordMarkers;
```

Methods Details

createMarkers()

```
Creates and returns a new validation object for this record, which allows for markers to be used outside the
validation flow.
Will overwrite the current markers if present.
Can be set to null again if you checked the problems, will also be set to null when a save was successful.
```

Returns

JSRecordMarkers A new validation object.

Supported Clients

SmartClient,WebClient,NGClient

Sample

```
var recordMarkers = record.createMarkers();
```

getChangedData()

```
Returns a JSDataSet with outstanding (not saved) changed data of this record.
column1 is the column name, colum2 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.

Supported Clients

SmartClient,WebClient,NGClient

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()

```
Returns the records datasource string.
```

Returns

String The datasource string of this record.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

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

getPKs()

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

Returns

Array an Array with the pk values.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

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

hasChangedData()

Returns true if the current record has outstanding/changed data.

Returns

Boolean true if the current record has outstanding/changed data.

Supported Clients

SmartClient,WebClient,NGClient

Sample

```
var hasChanged = record.hasChangedData();
```

isEditing()

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

This will not check if the record doesn't really have any changes, it just returns the edit state.
So this can return true but databaseManager.getEditedRecord() will not return this record because that
call will check if the record has really any changed values compared to the stored database values.
Record can be in edit mode without changes when some field is focused (so edit is started) but no changes are
done yet
or when changes were done in such a way that record data is the same as database data.

Returns

Boolean a boolean when in edit.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

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

isNew()

Returns true if the current record is a new record or false otherwise. New record means not saved to database.

Returns

Boolean true if the current record is a new record, false otherwise;

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var isNew = foundset.getSelectedRecord().isNew();
```

isRelatedFoundSetLoaded(relationName)

Returns true or false if the related foundset is already loaded. Will not load the related foundset.

Parameters

String relationName name of the relation to check for

Returns

Boolean a boolean when loaded.

Supported Clients

SmartClient,WebClient,NGClient

Sample

```
var isLoaded = foundset.getSelectedRecord().isRelatedFoundSetLoaded(relationName)
```

revertChanges()

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

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

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