

[Home](#) x [Install](#) x [Full Index](#) x [Tutorial](#) x [EnumerateControls](#) x [FindOutTableExists](#) x [UseVariablesInSQL](#) x [CreateRecordsetFrom](#) x [AddRecordToRecordset](#) x [CountRecordsRecordset](#) x [LimitsRecordset](#) x [MixAccess2baseAndUNO](#) ! [User's Guide](#) x [AllForms](#) x [DatabaseWindow](#) x [ShortcutNotationMore](#) x [DLookupSamples](#) x [CalculatedField](#) x [MultiSelectListboxSelectForm](#) x [FillAutoControlValue](#) x [CarryToNewRecord](#) x [BrowseThruControls](#) x [TipTextForLongValues](#) x [AskBeforeSaving](#) x [Sync2Combos](#) x [ZoomOnImage](#) x [AddAllToBox](#) x [KeepFormsSynchro](#) x [SelectListboxOnFirstLetters](#) x [MoveItemsBetweenListboxes](#) x [SimulateTabbed](#) x [SearchStandalone](#) x [CalculatorDialog](#) x [ExploreTables](#) x [ExtractDataTable](#) x [FindPositionRecordset](#) x [DMedian function](#) x [DPercentile](#) x [ImportImages](#) x [ExportImages](#) x [CrossTabQuery](#) x [DbaccessFromCalc](#) x [Standalone Forms](#) x [Add](#) x [AddItem](#) x [AddNew](#) x [CancelUpdate](#) x [Clone](#) x [Close \(method\)](#) x [CloseAllRecordsets](#) x [CloseConnection](#) x [CreateField](#) x [CreateQueryDef](#) x [CreateTableDef](#) x [CurrentDb](#) x [Delete](#) x [Delete \(table-query\)](#) x [Edit](#) x [EndExecute](#) x [Execute \(commandbarcontrol\)](#) x [Execute \(dialog\)](#) x [Execute \(query\)](#) x [getProperty](#) x [GetRows](#) x [hasProperty](#) x [Move](#) x [Move \(recordset\)](#) x [OpenConnection](#) x [OpenDatabase](#) x [OpenRecordset](#) x

OpenRecordset

Creates a new **Recordset** object and appends it to the **Recordsets** collection of the concerned **database** object.

tags:
Methods

Applies to ...

Object	Description
Database	The representation of a database.
TableDef	The representation of a database table.
QueryDef	The representation of a database query.
Recordset	The representation of another recordset.

Syntax

For a database object:

```
Set recordset = database.Openrecordset (source, type, option, lockedit)
```

For tabledef, querydef and recordset objects:

```

Set recordset = tabledef.Openrecordset (type, option, lockedit)
Set recordset = querydef.Openrecordset (type, option, lockedit)
Set recordset = recordset.Openrecordset (type, option, lockedit)

```

Arguments - Returned value

Argument	Type	Optional	Description	Returned value
source	String		A table name, a query name, or an SQL statement that returns records	A recordset object
type	Integer Long	Y	If the argument is present its only allowed value = <i>dbOpenForwardOnly</i> . Forces one-directional browsing of the records.	
option	Integer Long	Y	If the argument is present its only allowed value = <i>dbSQLPassThrough</i> . Forces escape substitution before sending the SQL statement to the database.	
lockedit	Integer Long	Y	If the argument is present its only allowed value = <i>dbReadOnly</i> . Forces dirty read and prevents from database updates.	

Remarks

- A recordset object MUST be closed by the **close** method as soon as it becomes useless to avoid eventual record

(dead)locks. To close all open recordsets at once, use the **CloseAllRecordsets** method.

- If the *OpenRecordset* method is applied to a *database* object and the *source* argument is not a SQL statement then the *source* is presumed to contain at first a TABLE name and secondly a QUERY name.
- To include the constants in your own code, copy and paste next lines:

```
Const dbOpenForwardOnly = 8
Const dbSQLPassThrough = 64
Const dbReadOnly = 4
```

- To access the recordset linked to a **form** or a **subform** use the **recordset** property.

Error messages

Argument nr. ... [Value = '...'] is invalid
SQL error, SQL statement = '...'
Table/Query '...' not found

See also

Close

CloseAllRecordsets

Filter

Recordset object

Recordset property

TableDefs

TableDef

QueryDefs

QueryDef

Example

Create a new recordset on table *Invoices* and count the records (not the most efficient way to do it ... !)

```
Const dbReadOnly = 4
Dim orsRecords As Object, lCount As Long
    Set orsRecords = Application.CurrentDb().OpenRecordset("INVOICES", , , dbReadOnly)
    lCount = 0
    With orsRecords
        If Not .BOF Then ' An empty recordset has both .BOF and
            Do While Not .EOF
                lCount = lCount + 1
                .MoveNext
            Loop
        End If
        .mClose()
    End With
Print "Number of records = " & lCount
```

Bookmark this page » » [OpenRecordset](#)