Recordset

A *Recordset* object represents the records in a base table or the records that result from running a query. A cursor technique allows to walk through the records of the recordset.

tags:
Objects

Functions returning a recordset object

Parent object	Method	Type
Database	**Recordsets**	**Collection**
Database	**OpenRecordset**	
TableDef		
QueryDef		
Recordset	**OpenRecordset**	
	Clone	
Form	**Recordset**	
SubForm		

Properties

Property	Type	Read only	Description or UNO object
AbsolutePosition			Sets or returns the relative record number of a *Recordset*.
BOF		Y	Specifies if the recordset cursor is placed before the first record.
Bookmark			Sets or returns a bookmark that uniquely identifies the current record in a *Recordset* object.
Bookmarkable		Y	Returns a value that indicates whether a *Recordset* object supports bookmarks.
EditMode		Y	Indicates the state of editing for the current record.
EOF		Y	Specifies if the recordset cursor is placed after the last record.
Filter			Specifies a selection to be set on the current recordset before opening with **OpenRecordset** a new derived recordset.
Name		Y	Specifies the name assigned by Access2Base to the recordset.
ObjectType		Y	Returns "RECORDSET".
RecordCount		Y	Returns the number of records of the recordset.

RowSet	UNO	Y	com.sun.star.comp.dba.ORowSet

Methods

Method	Description
AddNew	Initiate the creation of a new record.
CancelUpdate	Cancel on-going updates.
Clone	Create a duplicate *Recordset* object that refers to the current one.
Close	Close the *Recordset* object. The object is not usable anymore.
Delete	Delete the current record.
Edit	Initiate the update of fields in the current record.
Fields	Access the collection of fields of the recordset.
getProperty	Return the value of the given property.
GetRows	Retrieve multiple rows at once.
hasProperty	Return True if the Recordset has the given property.
Move	Walk through the recordset.
MoveFirst	
MoveLast	
MoveNext	
MovePrevious	
OpenRecordset	derive a new *Recordset* object from the current one.
setProperty	Return True if the given property could be set.
Update	Confirm on-going updates.

Remarks

- A new *Recordset* object is automatically added to the **Recordsets** collection when you open the object, and is automatically removed when you close it.
- You can create as many *Recordset* object variables as needed. Different *Recordset* objects can access the same tables, queries, and fields without conflicting.
- When you create a *Recordset* object, the current record is positioned to the first record if there are any records. If there are no records, the **RecordCount** property setting is 0, and the **BOF** and **EOF** property settings are `True`.
- Information about the structure of a base table, such as the names and data types of each **Field** object, is contained in a **TableDef** object.

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 » »