

[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 [OptionGroup \(Method\)](#) x [ReadAllBytes](#) x  
[ReadAllText](#) x [Refresh](#) x [Remove](#) x [RemoveAll](#) x [RemoveItem](#) x [Requery](#) x [Reset](#) x [RunSQL](#) x  
[SetFocus](#) x [setProperty](#) x [Start](#) x [Terminate](#) x [Update](#) x [WriteAllBytes](#) x [WriteAllText](#) x [Methods](#) x  
[Objects](#) x [Application](#) x [CommandBar](#) x [CommandBarControl](#) x [Control](#) x [Database](#) x [Dialog](#) x  
[DoCmd](#) x [Event](#) x [Field](#) x

## Field

A **field object** describes one of the fields of a **table**, a **query** or a **recordset**. Each field will be retrieved as a member of the **Fields collection** of its corresponding parent.

tags:  
Objects

### Functions returning a field object

Parent object	Function	Type	Arguments	Short example
<b>Recordset</b> <b>QueryDef</b> <b>TableDef</b>	<b>Fields</b>	<b>Collection</b>	None Integer, Long String	<code>Application.CurrentDb().TableDefs("myTable").Fields("myField")</code> returns an object of type <b>Field</b> referring to the <code>myField</code> field in the <code>myTable</code> table.

### Field types

The distinct field types can be recognized thru the use either of the **DataType** (long), the **DbType** (integer) or the **TypeName** (string) properties. The *DbType* property is proposed in *Access2Base* for compatibility with *MSAccess*.

See the correspondence table below.

DataType	DbType	TypeName
com.sun.star.sdbc.DataType.BIT	<i>dbUndefined</i>	BIT
com.sun.star.sdbc.DataType.BOOLEAN	dbBoolean	BOOLEAN
com.sun.star.sdbc.DataType.TINYINT	dbInteger	TINYINT
com.sun.star.sdbc.DataType.SMALLINT	dbLong	SMALLINT
com.sun.star.sdbc.DataType.INTEGER	dbLong	INTEGER
com.sun.star.sdbc.DataType.BIGINT	dbBigInt	BIGINT
com.sun.star.sdbc.DataType.FLOAT	dbFloat	FLOAT
com.sun.star.sdbc.DataType.REAL	dbSingle	REAL
com.sun.star.sdbc.DataType.DOUBLE	dbDouble	DOUBLE
com.sun.star.sdbc.DataType.NUMERIC	dbNumeric	NUMERIC
com.sun.star.sdbc.DataType.DECIMAL	dbDecimal	DECIMAL
com.sun.star.sdbc.DataType.CHAR	dbtext	CHAR
com.sun.star.sdbc.DataType.VARCHAR	dbChar	VARCHAR
com.sun.star.sdbc.DataType.LONGVARCHAR	dbMemo	LONGVARCHAR
com.sun.star.sdbc.DataType.DATE	dbDate	DATE
com.sun.star.sdbc.DataType.TIME	dbTime	TIME

com.sun.star.sdbc.DataType.TIMESTAMP	dbTimeStamp	TIMESTAMP
com.sun.star.sdbc.DataType.BINARY	dbBinary	BINARY
com.sun.star.sdbc.DataType.VARBINARY	dbVarBinary	VARBINARY
com.sun.star.sdbc.DataType.LONGVARBINARY	dbLongBinary	LONGVARBINARY
com.sun.star.sdbc.DataType.CLOB	dbUndefined	CLOB
com.sun.star.sdbc.DataType.BLOB	dbUndefined	BLOB

## Properties

Property	Type	Read only	Description or UNO object
<b>DataType</b>		Y	Specifies the <i>AOO/LibO</i> type of the data as an integer value
<b>DbType</b>		Y	Specifies the <i>MsAccess</i> type of the data as an integer value
<b>DefaultValue</b>			Returns the value stored in a new record
<b>Description</b>			A summary description of the field
<b>FieldSize</b>		Y	Returns the number of bytes used in the database of a Memo or Long Binary field
<b>Name</b>		Y	The exact name of the field
<b>ObjectType</b>		Y	Always "FIELD"
<b>Size</b>		Y	The maximum size of the field
<b>SourceTable</b>		Y	Indicates the name of the table that is the original source of the data for the field
<b>TypeName</b>		Y	Specifies the <i>AOO/LibO</i> type of the data as a string
<b>Value</b>			The value stored or to be stored in the field
Column	UNO	Y	com.sun.star.sdb.OTableColumnWrapper org.openoffice.comp.dbaccess.OQueryColumn com.sun.star.sdb.ODataColumn

## Methods

Method	Argument(s)	Return	Description
<b>getProperty</b>	property	Variant	Return the value of the given property.
<b>hasProperty</b>	property	Boolean	Return True if the Field has the given property.
<b>ReadAllBytes</b>	filename		Store a binary file in a binary field
<b>ReadAllText</b>	filename		Store a text file in a memo field
<b>setProperty</b>	property value		Return True if the given property could be set.
<b>WriteAllBytes</b>	filename		Copy a binary field into a file
<b>WriteAllText</b>	filename		Copy a memo field into a file

## Remarks

## Example

List all fields of a table with their types

```
Dim i As Integer, oTable As Object, oField As Object
Set oTable = Application.CurrentDb().TableDefs("AllTypes")
For i = 0 To oTable.Fields().Count - 1
    Set oField = oTable.Fields(i)
    DebugPrint oField.Name, oField.DataType, oField.DbType, oField.TypeName
Next i
```

