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

ExploreTables

(Q) How can I list table and field names, and use them in my code ?

tags:
HowTo

(R) We will present 2 techniques. a generic one which uses the Access2Base API for exploring *TableDefs*, and a second, which is specific to HSQLDB, which runs SQL statements on the database schema.

An easy solution is to use the TableDefs and Fields collections and their objects.

Here a simple list of tables with their fields and types:

```
Sub ScanTables()

Dim oDatabase As Object, oTable As Object, oField As Object
Dim i As Integer, j As Integer

        Set oDatabase = Application.CurrentDb()
        With odatabase
                For i = 0 To .TableDefs.Count - 1
                        Set oTable = .TableDefs(i)
                        DebugPrint oTable.Name

                        For j = 0 To oTable.Fields.Count - 1
                                Set oField = oTable.Fields(j)
                                DebugPrint "", LongStr(oField.Name), LongStr(oField.TypeName),
                        Next j
                Next i
        End With

End Sub

Function LongStr(psString As String) As String
Const cstLength = 20
        LongStr = Left(psString & Space(cstLength), cstLength)
End Function
```

Note that the *LongStr* function is there only to have the output cleanly aligned in columns.
The output will be something like:

```
    Categories
         CategoryName             VARCHAR                  50
         Description              VARCHAR                  2147483647
         Picture                  BINARY                   2147483647
         CategoryID               BIGINT                   0
    Customers
         Address                  VARCHAR                  50
         City                     VARCHAR                  50
         CompanyName              VARCHAR                  50
         ContactName              VARCHAR                  50
         ContactTitle             VARCHAR                  50
         Country                  VARCHAR                  50
         CustomerID                VARCHAR                 50
         Fax                       VARCHAR                 50
         Phone                     VARCHAR                 50
         PostalCode                VARCHAR                 50
         Region                    VARCHAR                 50
    Employees
         Address                   VARCHAR                 50
 etc ...
```

Solution by exploring the database schema directly (HSQLDB only)

Not surprisingly reading the HSQLDB documentation will help understand the following examples.
Running next code

```
Sub ScanSchemaSQL()
Dim sSql As String
        sSql = "SELECT [TABLE_NAME],[COLUMN_NAME],[SYSTEM_COLUMNS].[TYPE_NAME],[COLUMN_SIZE] '
                        & "FROM [INFORMATION_SCHEMA].[SYSTEM_TABLES],[INFORMATION_SCHEMA].[SYS
                        & "WHERE [TABLE_SCHEM]='PUBLIC' AND [SYSTEM_COLUMNS].[TABLE_NAME]=[SYS
        OpenSQL(sSql, dbSQLPassThrough)
End Sub
```

produces the result displayed below:

```
TABLE_NAME              COLUMN_NAME             TYPE_NAME       COLUMN_SIZE
Categories              CategoryName            VARCHAR         50
Categories              Description             VARCHAR         2147483647
Categories              Picture                 BINARY          2147483647
Categories              CategoryID              BIGINT
Customers               Address                 VARCHAR         50
Customers               City                    VARCHAR         50
Customers               CompanyName             VARCHAR         50
Customers               ContactName             VARCHAR         50
Customers               ContactTitle            VARCHAR         50
Customers               Country                 VARCHAR         50
Customers               CustomerID              VARCHAR         50
Customers               Fax                     VARCHAR         50
Customers               Phone                   VARCHAR         50
Customers               PostalCode              VARCHAR         50
Customers               Region                  VARCHAR         50
Employees               Address                 VARCHAR         50
etc ...
```

The same results can also be explored programmatically:

```
Sub ScanSchema()

Dim oRecordset As Object, sSql As String, iNbFields As integer
        sSql = "SELECT [TABLE_NAME],[COLUMN_NAME],[SYSTEM_COLUMNS].[TYPE_NAME],[COLUMN_SIZE] '
                        & "FROM [INFORMATION_SCHEMA].[SYSTEM_TABLES],[INFORMATION_SCHEMA].[SYS
                        & "WHERE [TABLE_SCHEM]='PUBLIC' AND [SYSTEM_COLUMNS].[TABLE_NAME]=[SYS

        Set oRecordset = Application.CurrentDb().OpenRecordset(sSql, , dbSQLPassThrough, dbRea
        With oRecordset
                iNbFields = .Fields.Count
                Do While Not .EOF()
                        DebugPrint LongStr(.Fields("TABLE_NAME").Value) _
                                                , LongStr(.Fields("COLUMN_NAME").Value) _
                                                , LongStr(.Fields("TYPE_NAME").Value) _
                                                , .Fields("COLUMN_SIZE").Value
                        .MoveNext()
                Loop
        End With

End Sub
```

produces once again the same result as before ...

```
Categories                CategoryName              VARCHAR                   50
Categories                Description               VARCHAR                   2147483647
Categories                Picture                   BINARY                    2147483647
Categories                CategoryID                BIGINT                    [NULL]
Customers                 Address                   VARCHAR                   50
Customers                 City                      VARCHAR                   50
Customers                 CompanyName               VARCHAR                   50
Customers                 ContactName               VARCHAR                   50
Customers                 ContactTitle              VARCHAR                   50
Customers                 Country                   VARCHAR                   50
Customers                 CustomerID                VARCHAR                   50
Customers                 Fax                       VARCHAR                   50
Customers                 Phone                     VARCHAR                   50
Customers                 PostalCode                VARCHAR                   50
Customers                 Region                    VARCHAR                   50
Employees                 Address                   VARCHAR                   50
etc ...
```

See also

DebugPrint
Fields
OpenSQL
Recordsets
TableDefs

Refer to ...

Basic module	Form	Form event	Control	Control event	Comments
Dictionary					Run one of the proposed Subs. They do not need any form or control event.

Bookmark this page » » ExploreTables