

## Tabelle automatisch aktualisieren

LO BASE Tabelle 'Eigner' Formular 'EINGER-DateUP' Schaltfläche ruft Makro 'Sub EignerDATEup1(oEvent As Object) auf.

Ziel: in allen DS der Tabelle bestimmte Feldinhalte zu ergänzen (Textfeldinhalte in DATE-Felder übertragen).

Das gelingt derzeit mit einem, dem aktuellen Formular.

Aufgabe: Einbau einer Hauptschleife, die das nächste Formular (mit dem nächsten DS) lädt, um ohne weiteren Usereingriff alle DS der Tabelle zu aktualisieren.

Meine bisherigen Versuche waren erfolglos.

### Sub EignerDATEup1(oEvent As Object)

```
Dim oForm As Object
Dim oDatenquelle As Object
Dim oVerbindung As Object
Dim oSQL_Anweisung As Object
Dim stSql As String
Dim fieldCount As Integer
Dim fieldNames() As String
Dim fieldValues() As String
Dim cDatum As String
Dim cDatumFormatted As String
Dim dDateValue As Date
```

' On Error GoTo ErrorHandler ' Fehlerbehandlung aktivieren

' Verbindung zur Datenbank herstellen, falls nicht verbunden

oDatenquelle = ThisComponent.Parent.CurrentController

If NOT (oDatenquelle.isConnected()) Then

oDatenquelle.connect()

End If

oVerbindung = oDatenquelle.ActiveConnection()

If oVerbindung Is Nothing Then

MsgBox "873 Die Verbindung zur Datenbank konnte nicht hergestellt werden."

Exit Sub

End If

“ ANFANG“ ' Anfang der Hauptschleife?

' SQL-Anweisung Anfang

stSql = "UPDATE ""Eigner"" SET "

oForm = oEvent.Source.Model.Parent ' Formularreferenz erhalten

fieldCount = oForm.getColumns().getCount() ' Anzahl der Felder im Formular ermitteln

' Arrays für Feldnamen und Werte initialisieren

ReDim fieldNames(fieldCount - 1)

ReDim fieldValues(fieldCount - 1)

' Innere Schleife:

' Feldnamen und Werte je Datensatz aus dem Formular auslesen

For i = 0 To fieldCount - 1

```

fieldNames(i) = oForm.getColumns().getByIndex(i).Name
Select Case oForm.getColumns().getByIndex(i).Type
Case 12 ' Text [VARCHAR] "String"
    fieldValues(i) = "" & Replace(oForm.getString(i + 1), "", "") & ""
' Bereich der Datumsfelder (Strings) abfragen
If fieldNames(i) = "Geb-Datum" OR fieldNames(i) = "Geb2-Dat" OR fieldNames(i) = "Vertr-
Beginn" OR fieldNames(i) = "Vertr-End" Then
    cdatum = Mid(fieldValues(i), 2, 2) 'auf Inhalt untersuchen
    IF cdatum > "00" Then 'Datum für i+1 aufbereiten
        MsgBox "898 i " & i & " " & fieldNames(i) & " " & fieldValues(i)
        " das aktuelle Stringfeld TT.MM.JJ wird in den DATE-Wert JJJJ-MM-TT convertiert
        " und für daas DATE-Feld vorgemerken
        cdatum = ConvertToISOFormat(fieldValues(i)) ' Funktionsaufruf
        fieldValues(i+1) = cDatum ' ganz wichtig für das UPDAT von i+1
    End IF
End IF

Case 91 ' aus Datum TT.MM.JJJJ DATE JJJJ-MM-TT erzeugen
    fieldNames(i) = "" & oForm.getColumns().getByIndex(i).Name & "" ' doppelte
Anführungszeichen!
    ' MsgBox "934 (" & i & ") " & fieldValues(i)
    If cDatum < 1 Then
        fieldValues(i) = "NULL" ' Setze auf NULL, ohne Anführungszeichen.
    Else
        fieldValues(i) = "" & fieldvalues(i) & ""
    End If
' vorhandenes oder generiertes DATE einsetzen:
stSql = stSql & fieldNames(i) & " = " & fieldValues(i)
If i < fieldCount - 1 Then
    stSql = stSql & ", "
End If

End Select

Next i ' Schleife je Datensatz

'SQL-Anweisung fortsetzen
i = Len(stSql) ' Länge feststellen um das letzte Komma zu entfernen
' MsgBox "934 Len = " & i
stSql = MID(stSql, 1, i-2) ' letztes Komme entfernt

stSql = stSql & " WHERE ""ID"" = " & recordID ' Füge die WHERE-Klausel hinzu

' Führe die SQL-Anweisung aus
oSQL_Statement = oVerbindung.createStatement()
oSQL_Statement.executeUpdate(stSql)

MsgBox "Update für einen DS erfolgreich"

"" hier müsste die zweite Schleife das nächste Formular aufrufen (mit dem jeweils
nächsten) und an „ANFANG“ zurückkehren.

End Sub

```