Prüfung auf leeren Recordset
Versionen: Access 2013, 2010, 2007, 2003 und 2002/XP

Zu unserem Helpline-Beitrag in Access Weekly vom 15.11.2013 haben uns verschiedene Zuschriften erreicht, für die wir uns bedanken möchten. Unter anderem wurde die folgende alternative Funktion für Prüfung auf einen leeren Recordset vorgeschlagen:

Public Function HasRecords(ByVal rs As Object) As Boolean

  If Not rs Is Nothing Then
    HasRecords = Not (rs.BOF And rs.EOF)
  Else
    HasRecords = False
  End if

End Function

Diese Funktion können Sie einsetzen, wenn Sie ausschließlich mit internen Access-Tabellen oder mit verknüpften Tabellen arbeiten, auf die über von Microsoft zertifizierte ODBC-Treiber zugreifen. Sie ist etwas schneller und universeller im Einsatz, da sie keine Datensätze ausliest und sowohl DAO- als auch ADO-Recordsets abdeckt. Es ist jedoch zu beachten, dass diese Funktion beim Einsatz verknüpfter Tabellen, auf die über ältere oder nicht von Microsoft zertifizierte ODBC-Treiber zugegriffen wird, möglicherweise ein falsches Ergebnis liefert. Obwohl die Tabelle Datensätze enthält, melden "BOF" und "EOF" mitunter den Zustand "True". Setzen Sie in solchen Fällen die vorgestellte Konstruktion ein, die durch "MoveLast" und Abfrage der Eigenschaft "RecordCount" zwar etwas langsamer ist, aber in jedem Fall ein zuverlässiges Ergebnis liefert:

Dim lngRecCnt As Long

On Error Resume Next
Set rs = db.OpenRecordset(<Hier Ihr Ausdruck>)
rs.MoveLast
lngRecCnt = rs.RecordCount
If Err <> 0 Or lngRecCnt <= 0 Then 'Keine Datensaetze
  MsgBox "Keine Datensaetze..."
  Exit Sub
Else
  rs.MoveFirst
  While Not rs.EOF
    'Verarbeitung...
    rs.MoveNext
  Wend
End If