Excel VBA ADOxを利用して、接続元のAccessをバックグラウンドで起動して、別の接続先のAccessから所定のテーブルデータを取得するVBAを記述して下さい。記述するVBAプログラムの手順は、①バックグラウンドでデータを取得する接続元のAccessを起動する。②接続元のAccessから別の接続先のAccessデータベースに対してODBC接続を確立する。③接続先のAccessデータベースから所定のテーブルデータをSQLでデータを再構成する。④再構成したデータを接続元のAccessにテーブルデータとして取り込む。⑤取り込んだテーブルデータCSVファイルとして所定のローカルフォルダに保存する。⑥接続先のAccessを閉じる。⑦接続元のAccessを閉じる。

以下は、Excel VBA ADOxを使用して接続元のAccessをバックグラウンドで起動して、別の接続先のAccessから所定のテーブルデータを取得するVBAプログラムです。手順は以下の通りです。

接続元のAccessをバックグラウンドで起動する。

接続元のAccessから別の接続先のAccessデータベースに対してODBC接続を確立する。

接続先のAccessデータベースから所定のテーブルデータをSQLでデータを再構成する。

再構成したデータを接続元のAccessにテーブルデータとして取り込む。

取り込んだテーブルデータをCSVファイルとして所定のローカルフォルダに保存する。

接続先のAccessを閉じる。

接続元のAccessを閉じる。

以下がVBAプログラムのコードです。

Sub GetTableDataFromAccess()

' 接続元のAccessファイルパス

Const srcDBPath As String = "C:\path\to\source\database.accdb"

' 接続先のAccessファイルパス

Const destDBPath As String = "C:\path\to\destination\database.accdb"

' 取得するテーブル名

Const tableName As String = "Table1"

' CSVファイルの保存先

Const csvPath As String = "C:\path\to\output\table1.csv"

' 接続元のAccessをバックグラウンドで起動する

Dim accessApp As Object

Set accessApp = CreateObject("Access.Application")

accessApp.Visible = False

accessApp.OpenCurrentDatabase srcDBPath

' 接続先のAccessデータベースに対してODBC接続を確立する

Dim cn As Object

Set cn = CreateObject("ADODB.Connection")

cn.Open "Driver={Microsoft Access Driver (\*.mdb, \*.accdb)};DBQ=" & destDBPath

' SQL文を作成する

Dim sql As String

sql = "SELECT \* FROM " & tableName

' SQL文を実行し、再構成したデータを取得する

Dim rs As Object

Set rs = CreateObject("ADODB.Recordset")

rs.Open sql, cn

' 接続元のAccessにテーブルデータとして取り込む

accessApp.DoCmd.TransferDatabase acExport, "Microsoft Access", destDBPath, acTable, tableName, tableName

' CSVファイルとして保存する

DoCmd.TransferText acExportDelim, , tableName, csvPath, True

' 接続先のAccessを閉じる

rs.Close

cn.Close

' 接続元のAccessを閉じる

accessApp.Quit

Set accessApp = Nothing

End Sub

Excel VBA ADOxを利用して、接続元のAccessをバックグラウンドで起動して、別の接続先のAccessから所定のテーブルデータを取得するVBAを記述して下さい。記述するVBAプログラムの手順は、①バックグラウンドでデータを取得する接続元のAccessを起動する。②接続元のAccessから別の接続先のAccessデータベースに対してODBC接続を確立する。③接続先のAccessデータベースから所定のテーブルデータをSQLでデータを再構成する。④再構成したデータを接続元のAccessにテーブルデータとして取り込む。⑤テーブルデータの左端に上段のデータに1から昇順に附番を行う。⑥附番後のテーブルデータをCSVファイルとして所定のローカルフォルダに保存する。⑦接続先のAccessを閉じる。⑧接続元のAccessを閉じる。

以下は、Excel VBA ADOxを利用して、接続元のAccessをバックグラウンドで起動して、別の接続先のAccessから所定のテーブルデータを取得するVBAのサンプルコードです。手順は、質問文の要望に沿って記述しています。

Sub GetDataFromRemoteAccessDatabase()

' 接続元のAccessファイルパスと接続先のAccessファイルパスを設定

Dim srcAccessFilePath As String

Dim destAccessFilePath As String

srcAccessFilePath = "C:\path\to\source\Database.accdb"

destAccessFilePath = "C:\path\to\destination\Database.accdb"

' 接続先のAccessデータベースに対するODBC接続文字列を設定

Dim destDbConnectionString As String

destDbConnectionString = "Driver={Microsoft Access Driver (\*.mdb, \*.accdb)};Dbq=" & destAccessFilePath & ";Uid=;Pwd=;"

' ADOオブジェクトを作成

Dim conn As ADODB.Connection

Dim rs As ADODB.Recordset

Set conn = New ADODB.Connection

Set rs = New ADODB.Recordset

' 接続元のAccessファイルをバックグラウンドで開く

Dim appAccess As Access.Application

Set appAccess = New Access.Application

appAccess.Visible = False

appAccess.OpenCurrentDatabase srcAccessFilePath

' 接続先のAccessデータベースに接続

conn.ConnectionString = destDbConnectionString

conn.Open

' SQL文を設定

Dim sql As String

sql = "SELECT \* FROM Table1"

' SQLを実行し、結果をレコードセットに格納

rs.Open sql, conn

' 接続元のAccessにテーブルデータとして取り込む

Dim tableName As String

tableName = "Table1"

appAccess.DoCmd.TransferDatabase acImport, "ODBC Database", destDbConnectionString, acTable, tableName, tableName

' 取り込んだテーブルデータの左端に1から昇順に附番を行う

Dim i As Long

For i = 0 To rs.RecordCount - 1

appAccess.CurrentDb.Execute "UPDATE Table1 SET ID=" & i + 1 & " WHERE ID=" & rs.Fields("ID").Value

rs.MoveNext

Next i

' 附番後のテーブルデータをCSVファイルとして保存する

Dim filePath As String

filePath = "C:\path\to\output\table.csv"

appAccess.DoCmd.TransferText acExportDelim, , "Table1", filePath, True

' 接続先のAccessを閉じる

conn.Close

' 接続元のAccessを閉じる

appAccess.CloseCurrentDatabase

appAccess.Quit

' 解放する

Set rs = Nothing

Set conn = Nothing

Set appAccess = Nothing

End Sub

Excel VBA ADOxを利用して、接続元のAccessをバックグラウンドで起動して、別の接続先のAccessから所定のテーブルデータを取得するVBAを記述して下さい。記述するVBAプログラムの手順は、①バックグラウンドでデータを取得する接続元のAccessを起動する。②接続元のAccessから別の接続先のAccessデータベースに対してODBC接続を確立する。③接続先のAccessデータベースから所定のテーブルデータをSQLでデータを再構成する。④再構成したデータを接続元のAccessにテーブルデータとして取り込む。⑤テーブルデータの左端に上段のデータに1から昇順に附番を行う。⑥附番後のテーブルデータをCSVファイルとして所定のローカルフォルダに保存する。⑦接続先のAccessを閉じる。⑧接続元のAccessを閉じる。

ChatGPT

9 / 9

以下は、Excel VBA ADOxを利用して、接続元のAccessをバックグラウンドで起動して、別の接続先のAccessから所定のテーブルデータを取得し、附番してCSVファイルとして保存するVBAプログラムの例です。

vb

Copy code

Sub ExportDataFromAccess()

' 接続先のAccessファイルのパスとファイル名

Const dbPath As String = "C:\path\to\your\access\db.accdb"

' 接続先のテーブル名

Const tableName As String = "table1"

' CSVファイルの保存先のフォルダーパス

Const saveFolder As String = "C:\path\to\your\save\folder\"

' ADOオブジェクトの作成

Dim con As New ADODB.Connection

Dim rs As New ADODB.Recordset

' 接続先のAccessファイルをバックグラウンドで開く

Dim appAccess As Object

Set appAccess = CreateObject("Access.Application")

appAccess.OpenCurrentDatabase dbPath, False

' 接続先のAccessデータベースに対してODBC接続を確立する

con.Open "Driver={Microsoft Access Driver (\*.mdb, \*.accdb)};Dbq=" & dbPath & ";Uid=;Pwd=;"

' 接続先のAccessデータベースから所定のテーブルデータをSQLで再構成する

rs.Open "SELECT \* FROM " & tableName, con

' レコードの附番を行う

Dim i As Long

i = 1

Do Until rs.EOF

rs.Fields(0).Value = i

i = i + 1

rs.MoveNext

Loop

' 附番後のテーブルデータをCSVファイルとして保存する

rs.Save saveFolder & "table1.csv", adPersistCSV

' レコードセットと接続を解放する

rs.Close

con.Close

' 接続先のAccessを閉じる

appAccess.Quit

End Sub

上記の例では、接続先のAccessファイルのパスとファイル名、接続先のテーブル名、CSVファイルの保存先のフォルダーパスをそれぞれ定数に設定しています。この例では、附番にレコードのIDを使用していますが、テーブルにIDが存在しない場合は、別のフィールドを使用する必要があります。また、ODBC接続の情報を適宜変更する必要があります。