Excel VBA ADOxを使用して、接続元のAccessを起動するVBAプログラムを記述して下さい。①データを取得する接続元のAccessを起動する。②起動した接続元のAccessから別の接続先のAccessデータベースに対してテーブルデータを取得するための、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の上段のデータに1から昇順にAccess上で附番を行う。⑥テーブルデータをCSVファイルとして所定の保存先に保存する。⑦接続先のAccessを閉じる。

以下は、Excel VBA ADOxを使用して、接続元のAccessを起動するVBAプログラムです。①から⑦までの手順を順番に実行します。

Sub ExportTableToCSV()

Dim con As ADODB.Connection

Dim rs As ADODB.Recordset

Dim sql As String

Dim csvPath As String

Dim appAccess As Access.Application

Dim dbPath As String

Dim tblName As String

'接続先のAccessデータベースのパスとテーブル名を指定する

dbPath = "C:\path\to\database.accdb"

tblName = "Table1"

'Accessをバックグラウンドで起動する

Set appAccess = New Access.Application

appAccess.Visible = False

appAccess.OpenCurrentDatabase dbPath

'接続を確立する

Set con = New ADODB.Connection

con.Open "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=" & dbPath & ";Persist Security Info=False;"

'SQLで再構成する

sql = "SELECT \* FROM " & tblName & ";"

Set rs = con.Execute(sql)

'テーブルデータを取り込む

With appAccess.CurrentDb.OpenRecordset(tblName)

Do While Not rs.EOF

.AddNew

For i = 0 To rs.Fields.Count - 1

.Fields(i).Value = rs.Fields(i).Value

Next i

.Update

rs.MoveNext

Loop

rs.Close

.Close

End With

'テーブルデータの左端の上段のデータに1から昇順にAccess上で附番を行う

Set rs = appAccess.CurrentDb.OpenRecordset(tblName)

rs.MoveFirst

Do While Not rs.EOF

rs.Edit

rs.Fields(0).Value = rs.AbsolutePosition

rs.Update

rs.MoveNext

Loop

rs.Close

'CSVファイルに書き出す

csvPath = "C:\path\to\csv\output.csv"

DoCmd.TransferText acExportDelim, , tblName, csvPath, True

'接続先のAccessを閉じる

appAccess.CloseCurrentDatabase

Set appAccess = Nothing

'接続を閉じる

con.Close

Set con = Nothing

End Sub

このプログラムでは、Accessをバックグラウンドで起動するために、New Access.ApplicationでAccessオブジェクトを作成し、Visible = Falseで非表示にしています。また、Accessを起動する前に、Data Sourceに接続先のAccessデータベースのパスを指定しています。

接続先のAccessデータベースに対してSQLを実行するために、New ADODB.ConnectionでConnectionオブジェクトを作成し、con.Openで接続を確立します。SQLの実行結果は、con.ExecuteでRecordsetオブジェクトに格納されます。

Excel VBA ADOxを使用して、接続元のAccessを起動するVBAプログラムを記述して下さい。①データを取得する接続元のAccessを起動する。②起動した接続元のAccessから別の接続先のAccessデータベースに対してテーブルデータを取得するための、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の上段のデータに1から昇順にAccess上で附番を行う。⑥テーブルデータをCSVファイルとして所定の保存先に保存する。⑦接続先のAccessを閉じる。

以下は、Excel VBA ADOxを使用して、接続元のAccessを起動するVBAプログラムの例です。このプログラムは、接続先のAccessデータベースからテーブルデータを再構成して、接続元のAccessに取り込んで、テーブルデータをCSVファイルに保存します。

Sub ImportDataFromAccess()

Dim conn As New ADODB.Connection

Dim rs As New ADODB.Recordset

Dim cmd As New ADODB.Command

Dim accApp As Object

' 接続元のAccessを起動する

Set accApp = CreateObject("Access.Application")

accApp.Visible = False

accApp.OpenCurrentDatabase "C:\path\to\source\Database.accdb"

' 接続先のAccessデータベースに接続する

conn.ConnectionString = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=C:\path\to\destination\Database.accdb;Persist Security Info=False;"

conn.Open

' 接続先のAccessデータベースからテーブルデータを取得する

cmd.ActiveConnection = conn

cmd.CommandType = adCmdText

cmd.CommandText = "SELECT \* FROM TableName"

rs.CursorLocation = adUseClient

rs.Open cmd

' 取得したテーブルデータを再構成して、接続元のAccessに取り込む

accApp.DoCmd.TransferDatabase acExport, "Microsoft Access", "C:\path\to\destination\Database.accdb", acTable, "TableName", "TableName", True, False

accApp.RunCommand acCmdCompactDatabase

' 附番を行う

accApp.RunCommand acCmdSelectAllRecords

accApp.RunCommand acCmdRecordsGoToNew

accApp.RunCommand acCmdSortAscending

accApp.RunCommand acCmdSelectRecord

accApp.RunCommand acCmdRecordsGoToNew

' テーブルデータをCSVファイルとして保存する

accApp.DoCmd.TransferText acExportDelim, , "TableName", "C:\path\to\CSVFile.csv", True

' 接続先のAccessを閉じる

conn.Close

' 接続元のAccessを閉じる

accApp.Quit

Set accApp = Nothing

End Sub

このプログラムでは、CreateObjectメソッドを使用して、接続元のAccessアプリケーションをバックグラウンドで起動しています。また、DoCmdオブジェクトを使用して、Accessの内部コマンドを呼び出しています。附番を行うためには、テーブルデータを選択してから、acCmdSortAscendingコマンドを使用して、昇順にソートし、acCmdSelectRecordコマンドを使用して、レコードを選択して附番を行います。最後に、TransferTextメソッドを使用して、CSVファイルとしてテーブルデータを

Excel VBA ADOxを使用して、接続元のAccessを起動するVBAプログラムを記述して下さい。①データを取得する接続元のAccessを起動する。②起動した接続元のAccessから別の接続先のAccessデータベースに対してテーブルデータを取得するための、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の上段のデータに1から昇順にAccess上で附番を行う。⑥テーブルデータをCSVファイルとして所定の保存先に保存する。⑦接続先のAccessを閉じる。

以下は、Excel VBA ADOxを使用して、接続元のAccessを起動するVBAプログラムの例です。この例では、接続元のAccessデータベースから別のAccessデータベースにある特定のテーブルを取得し、CSVファイルとして保存します。必要に応じて、データベース名、テーブル名、ファイル名、ファイルパスを変更してください。

Sub ExportTableToCSV()

Dim conn As New ADODB.Connection

Dim rs As New ADODB.Recordset

Dim cmd As New ADODB.Command

Dim csvFile As String

Dim tblName As String

Dim dbPath As String

Dim sql As String

' 接続元のAccessデータベースを開く

conn.Open "Provider=Microsoft.ACE.OLEDB.12.0;" & \_

"Data Source=C:\Users\UserName\Documents\MyDatabase.accdb;"

' 接続先のAccessデータベースのパスとテーブル名を指定する

tblName = "MyTable"

dbPath = "C:\Users\UserName\Documents\MyOtherDatabase.accdb"

' 接続先のAccessデータベースに接続する

conn.Open "Provider=Microsoft.ACE.OLEDB.12.0;" & \_

"Data Source=" & dbPath & ";"

' SQLクエリを構築する

sql = "SELECT \* FROM " & tblName

' コマンドオブジェクトを作成し、SQLを設定する

Set cmd.ActiveConnection = conn

cmd.CommandText = sql

' レコードセットオブジェクトを作成し、SQLを実行する

rs.Open cmd, , adOpenForwardOnly, adLockReadOnly

' テーブルの左端に昇順に番号を附番する

rs.MoveFirst

Dim i As Long

i = 1

Do While Not rs.EOF

rs.Fields(0).Value = i

rs.Update

i = i + 1

rs.MoveNext

Loop

' CSVファイルに書き出す

csvFile = "C:\Users\UserName\Documents\MyTable.csv"

rs.Save csvFile, adPersistCSV

' レコードセットを閉じる

rs.Close

' 接続先のAccessデータベースを閉じる

conn.Close

' 接続元のAccessデータベースを閉じる

conn.Close

End Sub