**Data-Acquisition**

As already outlined in the problem description, Excel is still the working horse in many companies. And, how could it be otherwise, the same companies that heavily relay on Excel use the classic among databases: relational databases. Thus, it is necessary to embed SQL statement into the VBA framework as a starting point. The table below shows a potential data format in an relations format (note that is possible to have missing values or duplicates):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **var name** | **var name** | **var name** | **var name** | **var name** |
| varchar() |  | float() | date() |  |
| varchar() | char() |  | date() | int() |
| varchar() | char() |  | date() |  |
| varchar() | char() | float() | date() | int() |
| varchar() | char() | float() | date() |  |
| varchar() | char() | float() | date() | int() |

Here is an example of a data base connection:

1. Fire up excel and hit Alt+F11
2. Select Microsoft Activex Data Objects Library
3. Create a sup procedure and start coding:
   1. Establish data base connection

Dim sSQLQry As String

Dim ReturnArray

Dim Conn As New ADODB.Connection

Dim mrs As New ADODB.Recordset

Dim DBPath As String, sconnect As String

DBPath = ThisWorkbook.FullName 'Refering the sameworkbook as Data Source

'You can provide the full path of your external file as shown below

'DBPath ="C:\InputData.xlsx"

sconnect = "Provider=MSDASQL.1;DSN=Excel Files;DBQ=" & DBPath & ";HDR=Yes';"

'If any issue with MSDASQL Provider, Try the Microsoft.Jet.OLEDB:

'sconnect = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & DBPath \_

& ";Extended Properties=""Excel 8.0;HDR=Yes;IMEX=1"";"

* 1. Connect to data base

Conn.Open sconnect

* 1. Specify your SQL query

sSQLSting = "SELECT \* From [Sheet1$]" ' Your SQL Statement (Table Name= Sheet Name=[Sheet1$])

* 1. Fetching data

mrs.Open sSQLSting, Conn

* 1. Define where you want to add you data

Sheet2.Range("A2").CopyFromRecordset mrs

* 1. Disconnect from data base

Close Recordset

mrs.Close

'Close Connection

Conn.Close

One common use case it to update the data basis periodically. This can be achieved by inserting the following code snippet:

ActiveWorkbook.Connections("test").Refresh 'Refresh only the connection named "test"

ActiveWorkbook.RefreshAll 'Refresh all connections