```
Option Explicit
' Demonstration des Zugriffs auf TM1 mit Hilfe des TM1Object-Klassenmoduls
' (C) INFOLOG GmbH 2010

Private Const Adminhost As String = "localhost"
Private Const servername As String = "local"
Private Const username As String = "admin"
Private Const password As String = ""

Private Sub TM1Test()
    Dim tm1 As TM1API
    Dim server As TM1Server
    Dim cube As TM1Cube
    Dim elemB As TM1Element
    Dim elemC As TM1Element
    Dim dimens(1 To 2) As TM1Dimension
    Dim subsets(1 To 2) As TM1Subset
    Dim view As TM1View
    Dim elems As TM1Elements
    Dim value1 As Variant
    Dim value2 As Variant
    Dim value3 As Variant

    ' Creating TM1API-Instance
    Set tm1 = New TM1API

    ' Setting adminhost
    tm1.Adminhost = Adminhost

    ' Login to server
    Set server = tm1.Servers(servername) ' Setting server
    server.Login username, password

    ' Creating two dimensions

    ' Creating new elementcollection
    Set elems = New TM1Elements

    ' Creating new base-element with an auto-generated name (e.g. "TempElem0001")
    elems.Create "", tm1BaseElement

    ' Creating a new dimension using this elements-Collection
    Set dimens(1) = server.Dimensions.Create("", elems)
    Set elems = Nothing
    Set elems = New TM1Elements

    ' Creating a base-element using a hardcoded name
    Set elemB = elems.Create("TestElemB", tm1BaseElement)
    ' Creating a new consolidated element
    Set elemC = elems.Create("", tm1ConsolidatedElement)

    ' Cumulating the base-element
    elemC.ElementComponents.Create elemB, tm1ConsolidatedElement
    Set elemC = Nothing
    Set elemB = Nothing

    ' Creating a string-element
    elems.Create "TestElemS", tm1StringElement

    Set dimens(2) = server.Dimensions.Create("", elems)
    Set elems = Nothing

    ' Creating a new cube
    Set cube = server.Cubes.Create("", dimens(1).Name, dimens(2).Name)

    ' Writing two cube-cell-values
    value1 = 12345.67
    cube.CellValue(dimens(1).Elements(1).Name, "TestElemB") = value1
    value2 = "TestString"
    cube.CellValue(dimens(1).Elements(1).Name, "TestElemS") = value2


    ' Reading a cubecellvalue
    value1 = cube.CellValue(dimens(1).Elements(1).Name, dimens(2).Elements(1).Name)

    ' Creating one subset per dimension
    Set subsets(1) = dimens(1).SubsetsPrivate.Create()
    subsets(1).FillWithAllElements
    Set subsets(2) = dimens(2).SubsetsPrivate.Create()
    subsets(2).FillWithAllElements
```

```
    ' Creating a view
    Set view = cube.ViewsPrivate.CreateToLoopByArray("", subsets())
    view.Save

    ' Calculating the view
    view.Calculate

    ' Reading a few view-cell-values
    value1 = view.CellValue(1, 3)
    value2 = view.CellValue(2, 3)
    value3 = view.CellValue(3, 3)
    Debug.Print "value 1: " + Replace$(CStr(value1), ",", ".")
    Debug.Print "value 2: " + Replace$(CStr(value2), ",", ".")
    Debug.Print "value 3: " + Replace$(CStr(value3), ",", ".")

    ' Deleting the view
    view.Delete
    Set view = Nothing

    ' Deleting the subsets
    subsets(1).Delete
    Set subsets(1) = Nothing
    subsets(2).Delete
    Set subsets(2) = Nothing

    ' Deleting the cube
    cube.Delete
    Set cube = Nothing

    ' Deleting the dimensions
    dimens (1).Delete
    Set dimens(1) = Nothing
    dimens (2).Delete
    Set dimens (2) = Nothing

    ' Logging out from server
    server.Logout
    Set server = Nothing

    'Cleaning up
    tm1.CloseAPI
    Set tm1 = Nothing
End Sub

Private Sub main()
    TM1Test
End Sub
```