



TWIAV – Tips & tricks

October 2, 2010 – Tip MB002: How to create a custom function in MapBasic?

How to create a custom function in MapBasic?

This document describes how to create a new function, a custom function in MapBasic.

As an example a function is created to check whether a table is open or not. This function can come in handy in many occasions – you do not want your app to crash when it tries to open a table which has been opened by the user already, do you? Or – equally annoying: you do not want your app to crash when it tries to close a table which has been closed by the user already, do you?

So, before performing any operation on a table, it is always a good idea to check the status of the table: opened or closed.

Below you will find the code of this function.

TIP: Make a description!

If you create a custom function which is going to be used a lot – by you, by your colleagues, in multiple projects, over a prolonged period of time – it is always good to make a description and make this description known to your fellow developers in the organization. This might save other people a lot of time: it avoids the same kind of function being created over and over again...:-)

At the end of this document you will find a description of the function **TableIsOpen()** in the style of the MapBasic Help.

Custom Function

The MapBasic language supports many different functions. Some are standard BASIC functions (for example, **Asc()**, **Format\$()**, **Val()**, etc.) and some are unique to MapInfo Professional and MapBasic (for example, **Distance()** and **ObjectGeography()**). MapBasic also lets you define custom functions. Once you have defined a custom function, you can call that function just as you can call any of MapBasic's standard functions.

The body of a custom function is defined within a **Function...End Function** construction, which is syntactically very similar to a **Sub...End Sub** construction. The general syntax of a **Function...End Function** construct is as follows:

```
Function function_name( parameters, if any ) As  
  data_type  
    statement list  
End Function
```

The function itself has a data type. This dictates which type of value (for example, Integer, Date, String) the function returns when called.

Within the body of the **Function...End Function** construction, the function name acts like a byreference parameter. A statement within the **Function...End Function** construction can assign a value to the function name; this is the value that MapBasic later returns to the function's caller.

Declaration

Before you can create a custom function, you first have to declare the function, i.e. to define the name and parameter list of the function.

You have to use the **Declare Function** statement to do so:

```
Declare Function TableIsOpen  
  (ByVal sTabName As String)  
As Logical
```

Please note:

This declaration should appear at the beginning of your program, before the first procedure (the **Main** procedure), together with the other declarations, includes and definitions.

Creation

```
' *****
Function TableIsOpen (
    ByVal sTabName As String)
As Logical

    Dim iCounter, iNumberOfTables As Integer

    iNumberOfTables = NumTables()
    If iNumberOfTables = 0 Then
        Goto no_tables_open
    End If

    For iCounter = 1 to iNumberOfTables
        If TableInfo(iCounter, TAB_INFO_NAME) = sTabName
        Then
            TableIsOpen = TRUE
            Exit Function
        End If
    Next

    no_tables_open:
    TableIsOpen = FALSE

End Function

' *****
```

TableIsOpen() function

Purpose

Checks whether a table is open or not.

Syntax

TableIsOpen(*tablename*)

tablename is a string representing a table name

Return Value

Logical

Description

The TableIsOpen() function compares a given string to the names of all the open tables.

If the string is equal to the name of an open table the function returns TRUE.

Example

```
Dim sTable as String

sTable = "FR_Départements2006"

If TableIsOpen(sTable) Then
    Map From sTable
End If
```