

Custom Functions stored in a database

Since we got this nice database for JavaScript to do custom functions (see blog article: [Custom Functions in JavaScript](#)), we further thought about doing this with FileMaker directly. Imaging to have your custom function in a central start database. Your solution opens, defines your own environment to store them and then loads all your custom functions from a table in the database. You may even have flags to define which are active vs. inactive, testing vs. production quality or maybe limited to admins. Once you have them loaded in memory, you can call them anywhere in any open file as long as FileMaker is running.

Here is our example database:

The screenshot shows the 'FileMaker Custom Functions' window. At the top, there's a toolbar with navigation and search icons. Below the toolbar, a blue header bar contains the title 'FileMaker Custom Functions' and a descriptive paragraph: 'We can store custom functions in a table and load them into memory and call them from everywhere using MBS functions. Multiple parameters possible and data types are preserved.' Below the header, the function 'Concat' is defined. It has a declaration 'Concat(Value1; Value2)' and a description 'A quick test function to concat two texts'. The parameters section shows two input fields, 'Value1' and 'Value2', with corresponding labels 'The first parameter' and 'The second parameter', and example values 'Hello' and 'World'. The body of the function is 'Value1 & Value2'. At the bottom, there's a 'Result' field and a 'SampleCall' field containing the MBS function call: 'MBS("FM.CF"; "CF"; "Concat"; Value1; Value2)'. Navigation buttons like 'Format', 'Check', 'Register Function', and 'Register and Try' are also visible.

Name	Concat									
Declaration	Concat(Value1; Value2)									
Description	A quick test function to concat two texts									
Parameters	<table border="1"><thead><tr><th>Value1</th><th>The first parameter</th><th>Hello</th></tr></thead><tbody><tr><td>Value2</td><td>The second parameter</td><td>World</td></tr><tr><td></td><td></td><td></td></tr></tbody></table>	Value1	The first parameter	Hello	Value2	The second parameter	World			
Value1	The first parameter	Hello								
Value2	The second parameter	World								
Body	Value1 & Value2									
Result										
SampleCall	MBS("FM.CF"; "CF"; "Concat"; Value1; Value2)									

As you see we can define functions with as many parameters as needed. Then you enter the body of the function with whatever code you have in your custom functions. We got a button for macOS to

format the calculation with our [Syntaxcoloring](#) functions. Then we got a button to check errors, which will let FileMaker parse your code with the parameter definition included, so we can find syntax errors. Sadly we can't tell where the error is exactly, but we may tell you if the syntax is okay or what error code we got.

Our start script set ups a global dictionary named "CF" in our example using [Dictionary.CreateNamed](#) function. Then you add functions there with [Dictionary.SetValueForKey](#) one by one or with [Dictionary.AddSQL](#) function as a block. Each function would have comments in front to name the parameters. We use "/// " & \$name there, e.g. "/// value".

Then later you can call the custom function with our [FM.CF](#) function. You pass the dictionary identifier, "CF" in our example. Then you pass the name of the function (case sensitive) and the parameters. As many parameters as you need, so you may define your function with 10 parameters, but only pass 2 or 3 depending on what you do.

The dictionary is global for all files and all solutions opened in FileMaker and can be initialized in FileMaker Pro, Server (Scripting + Web Direct) as well as FileMaker iOS SDK based applications. Since we use a dictionary identifier, you can have different sets of custom functions in different namespaces.

Please try the [JS.CF](#) function with [MBS FileMaker Plugin](#) in version 11.3pr3 or later.