

# **Storing data in the FileMaker schema**

Since we all love to make clones for using the data migration tool, we like to pass through some data with the clone. But a clone can't have records!

If we like to pass data in the clone like a version number, we need to carry it inside the schema. This happens by putting the text into the default value of a field. Like having a special table for global settings and there a version field with the version number as default value.

## **Add field with default**

You can modify the schema on FileMaker Pro, if it is not busy. So we usually do it on idle time with our [FM.ExecuteFileSQLOnIdle](#) function and run it in a script pause. The SQL command uses ALTER command with the ADD variant to add the field and we pass the default value in single quotes after the DEFAULT keyword. After the script pause, you can inspect error status and see whether the command was successful. Here is a sample script:

```
Set Variable [ $r ; Value: MBS("FM.ExecuteFileSQLOnIdle";"ALTER TABLE
\"Test\" ADD \"Version\" VARCHAR DEFAULT '1.2' "; Get(FileName)) ]
Pause/Resume Script [ Duration (seconds): ,2 Duration (seconds): .2 ]
Set Variable [ $ErrorCode ; Value: MBS("FM.ExecuteSQL.LastError") ]
Set Variable [ $Message ; Value: MBS("FM.ExecuteSQL.LastErrorMessage") ]
#
If [ $ErrorCode = 0 ]
    Show Custom Dialog [ "Added field" ; "Okay" ]
Else
    Show Custom Dialog [ "Added field" ; $Message ]
End If
```

## **Query the value**

Now to get the default value, we temporarily create a new record, check the Version field and then discard the new record like in this sample script:

```
New Record/Request
Show Custom Dialog [ Test::Version ]
Revert Record/Request [ With dialog: Off Off ]
```

Doing this in SQL would need an INSERT statement, a way to find the new record and a DELETE statement. Much easier to just go on the layout and make a temporary record without committing it to the database. There is currently no way to query the default value of a field in FileMaker directly.

## Update field with default

You can use SQL on idle in FileMaker Pro using the ALTER TABLE function with the "SET DEFAULT" command to put a new text in the default value. Again we need to have a script pause to give time to run this command like in this sample script:

```
Set Variable [ $r ; Value: MBS("FM.ExecuteFileSQLOnIdle";"ALTER TABLE
\"Test\" ALTER \"Version\" SET DEFAULT '1.3' "; Get(FileName)) ]
Pause/Resume Script [ Duration (seconds): ,2 Duration (seconds): .2 ]
Set Variable [ $ErrorCode ; Value: MBS("FM.ExecuteSQL.LastError") ]
Set Variable [ $Message ; Value: MBS("FM.ExecuteSQL.LastErrorMessage") ]
#
If [ $ErrorCode = 0 ]
    Show Custom Dialog [ "Added field" ; "Okay" ]
Else
    Show Custom Dialog [ "Added field" ; $Message ]
End If
```

## Perform on Server

You can do the same SQL commands on the server, but there you use the regular FM.ExecuteFileSQL function. Please use MBS Plugin 14.2 as we needed to change the plugin to make this work:

```
Set Variable [ $Message ; Value: MBS("FM.ExecuteFileSQL"; Get(FileName);
"ALTER TABLE \"Test\" ADD \"Version\" VARCHAR DEFAULT '1.2' ") ]
and
Set Variable [ $Message ; Value: MBS("FM.ExecuteFileSQL"; Get(FileName);
"ALTER TABLE \"Test\" ALTER \"Version\" SET DEFAULT '1.3' ") ]
```

## ModCount

You can query the modification count for the field, so you know whether it was updated.

```
Set Variable [ $Message ; Value: MBS("FM.ExecuteFileSQL"; Get(FileName);
"SELECT ModCount FROM FileMaker_FIELDS WHERE TableName = 'Test' AND
FieldName = 'Version' ") ]
Show Custom Dialog [ "Modcount" ; $Message ]
```

Thanks to Nils Waldherr for bringing this idea to me. He made a sample file for you and published it on [github.com/fmgarage/ft-buildinfo](https://github.com/fmgarage/ft-buildinfo). Enjoy!  
Please try and let us know whether you have questions.