Record Creation Script Trigger

Today we want to share two ways to trigger scripts via MBS Plugin on record creation.
There has been a request to have

Auto Enter Script Trigger

First you can define a field TriggerScriptOnCreate with an auto calculation like this:

MBS("FM.RunScript"; "Contacts"; "RecordCreated"; Get(RecordID))

We set the option to not replace existing values. This calculation now runs on every record creation in FileMaker Pro. The plugin function returns OK, which is written to the record.
The script will now run whenever you create a new record or duplicate one as the calculation runs. We pass the record ID so the script could check the record.

Timer Script Trigger

Other people may create records on the same database hosted on a server. So you may want to regularly check whether there are new records and we just automated it via counting records via SQL to trigger a script when the count changes. Here is the sample script:

StartRecordCreatedTrigger

Set Variable [$count ; Value: MBS("FM.ExecuteFileSQL"; "Contacts"; "SELECT Count(*) FROM Contacts"; 9; 13)]
If [MBS("IsError")]
    Show Custom Dialog ["SQL Error?" ; $$LastSQLCount]
Else
    Set Variable [$$LastSQLCount ; Value: $count]
    # free old if still runs
    Set Variable [$r ; Value: If(Length($$RecordCreatedScheduleRef) > 0 ; MBS("Schedule.Release"; $$RecordCreatedScheduleRef); "")]
    # launch new evaluate schedule
    Set Variable [$$RecordCreatedScheduleRef ; Value: MBS("Schedule.EvaluateAfterDelay"; 60; "Let([$SQLCount = MBS(\"FM.ExecuteFileSQL\"; \"Contacts\"; \"SELECT Count(*) FROM Contacts\"; 9; 13); $e = MBS(\"IsError\"); $r = Wenn($SQLCount ≠ $$LastSQLCount; MBS(\"FM.RunScript\"; \"Contacts\"; \"RecordCr…]
End If

As you see we query an initial count number and put it in a global variable. We schedule an evaluate and let's take a look on the expression without the backslashes needed:

Let([
    $SQLCount = **MBS**("FM.ExecuteFileSQL"; "Contacts"; "SELECT Count(*) FROM Contacts"; 9; 13);
    $e = **MBS**("IsError");
    $r = Wenn($e = 0 AND $SQLCount ≠ $$LastSQLCount; **MBS**("FM.RunScript"; "Contacts"; "RecordCreated"));
    $$LastSQLCount = Wenn($e; $$LastSQLCount; $SQLCount)
]; "")

As you see we query again the count of records via SQL and if that number changes, trigger the script and remember the new count. If an error is reported from SQL, we ignore this for now as the file may be closed currently?

To stop this later, we can release the schedule and have the plugin stop it:

StopRecordCreatedTrigger

free old if still runs
Set Variable [$r ; Value: If(Length($$RecordCreatedScheduleRef) > 0 ;
MBS("Schedule.Release"; $$RecordCreatedScheduleRef); "")]
Set Variable [$$RecordCreatedScheduleRef ; Value: ""]

As you see we just call Schedule.Release function to release if the variable is set and then clear it.
By using the MBS Plugin to schedule this, we can do the Evaluate even while a script is running already. Your scripts can use OnTimer if needed as we don't need that.

Let us know if you have comments or questions.