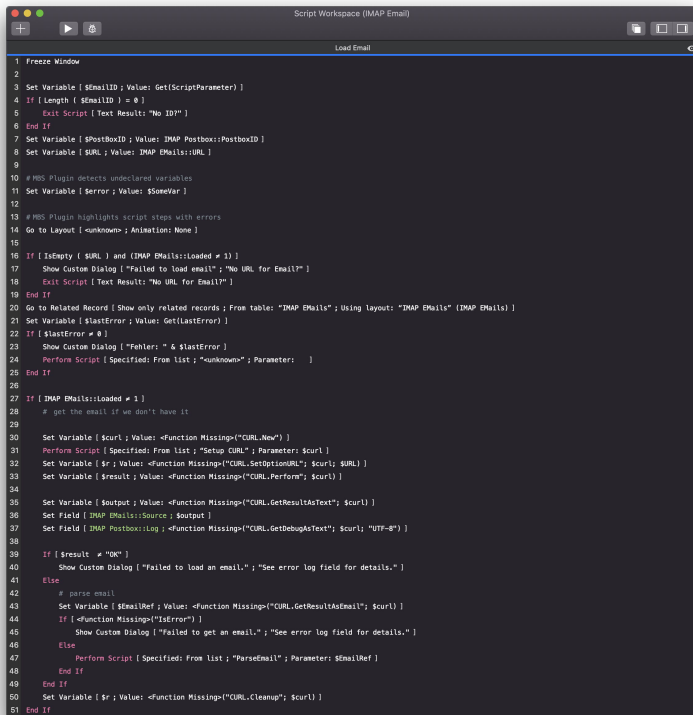


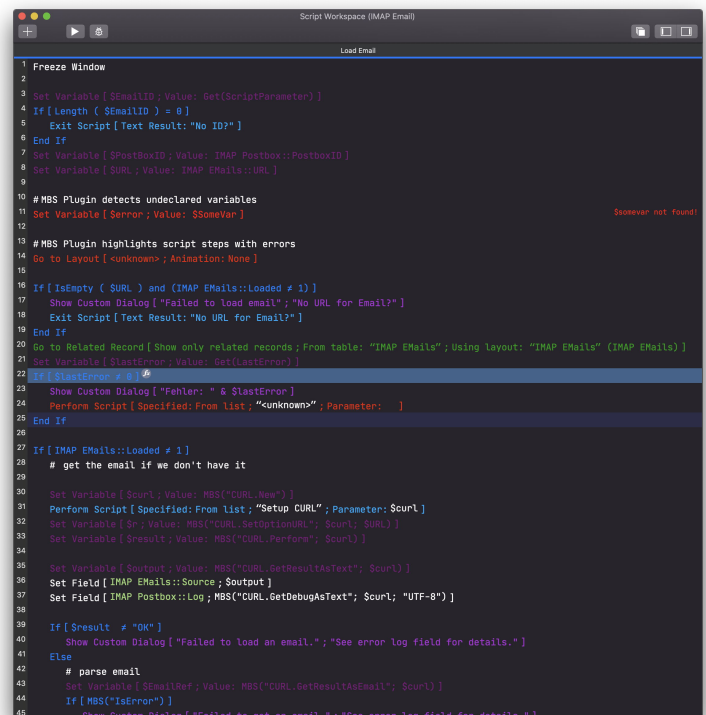
Dark Mode in FileMaker

With latest [MBS FileMaker Plugin](#) 10.3 we tuned our dark mode support for FileMaker. While Filemaker's Script Workspace comes with some syntax highlighting features built-in, our [MBS FileMaker Plugin](#) can add more for macOS users:



The screenshot shows the FileMaker Script Workspace in light mode. The script is titled 'Load Email' and contains 51 lines of code. The code is not syntax-highlighted, appearing in a standard black font on a white background. The interface includes a toolbar at the top with icons for running, freezing, and saving the script, and a status bar at the bottom.

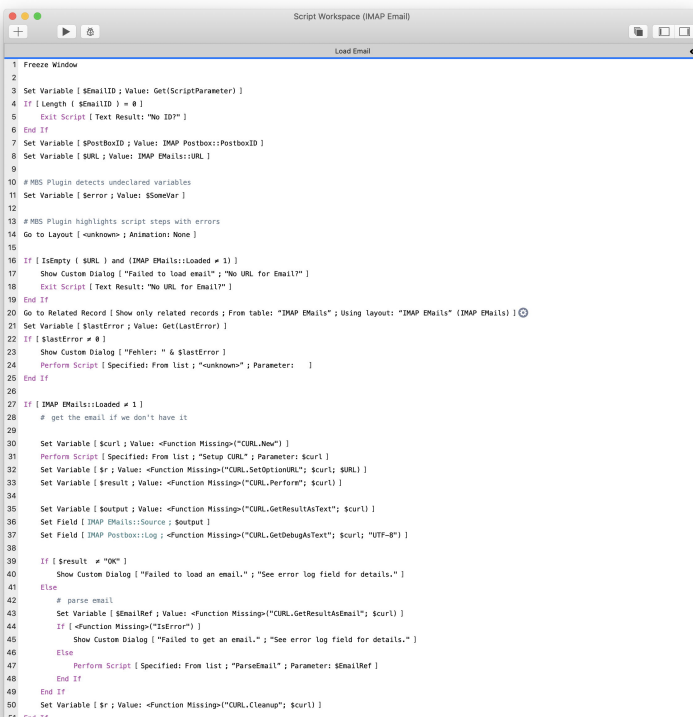
```
1 Freeze Window
2
3 Set Variable [ $EmailID ; Value: Get( ScriptParameter ) ]
4 If [ Length ( $EmailID ) = 0 ]
5     Exit Script [ Text Result: "No ID?" ]
6 End If
7 Set Variable [ $PostboxID ; Value: IMAP Postbox::PostboxID ]
8 Set Variable [ $URL ; Value: IMAP Emails::URL ]
9
10 #MBS Plugin detects undeclared variables
11 Set Variable [ $error ; Value: $SomeVar ]
12
13 #MBS Plugin highlights script steps with errors
14 Go to Layout [ <unknown> ; Animation: None ]
15
16 If [ IsEmpty ( $URL ) and ( IMAP Emails::Loaded ≠ 1 ) ]
17     Show Custom Dialog [ "Failed to load email"; "No URL for Email?" ]
18     Exit Script [ Text Result: "No URL for Email?" ]
19 End If
20 Go to Related Record [ Show only related records ; From table: "IMAP Emails" ; Using layout: "IMAP Emails" (IMAP Emails) ]
21 Set Variable [ $LastError ; Value: Get( LastError ) ]
22 If [ $LastError ≠ 0 ]
23     Show Custom Dialog [ "Fehler: " & $LastError ]
24     Perform Script [ Specified: From List ; "unknown" ; Parameter: ]
25 End If
26
27 If [ IMAP Emails::Loaded ≠ 1 ]
28     # get the email if we don't have it
29
30     Set Variable [ $curl ; Value: <function Missing>("CURL.New") ]
31     Perform Script [ Specified: From List ; "Setup CURL" ; Parameter: $curl ]
32     Set Variable [ $r ; Value: <function Missing>("CURL.SetOptionURL"; $curl; $URL) ]
33     Set Variable [ $result ; Value: <function Missing>("CURL.Perform"; $curl) ]
34
35     Set Variable [ $output ; Value: <function Missing>("CURL.GetResultAsText"; $curl) ]
36     Set Field [ IMAP Emails::Source ; $output ]
37     Set Field [ IMAP Postbox::Log ; <function Missing>("CURL.GetDebugAsText"; $curl; "UTF-8") ]
38
39     If [ $result ≠ "OK" ]
40         Show Custom Dialog [ "Failed to load an email."; "See error log field for details." ]
41     Else
42         # parse email
43         Set Variable [ $EmailRef ; Value: <function Missing>("CURL.GetResultAsEmail"; $curl) ]
44         If [ <function Missing>("IsError") ]
45             Show Custom Dialog [ "Failed to get an email."; "See error log field for details." ]
46         Else
47             Perform Script [ Specified: From List ; "ParseEmail"; Parameter: $EmailRef ]
48         End If
49     End If
50     Set Variable [ $r ; Value: <function Missing>("CURL.Cleanup"; $curl) ]
51 End If
```



The screenshot shows the FileMaker Script Workspace in dark mode with the MBS FileMaker Plugin installed. The script is titled 'Load Email' and contains 46 lines of code. The code is syntax-highlighted, with different colors used for keywords, strings, and comments. The interface includes a toolbar at the top and a status bar at the bottom. A red error message 'Somevar not found!' is visible on the right side of the script.

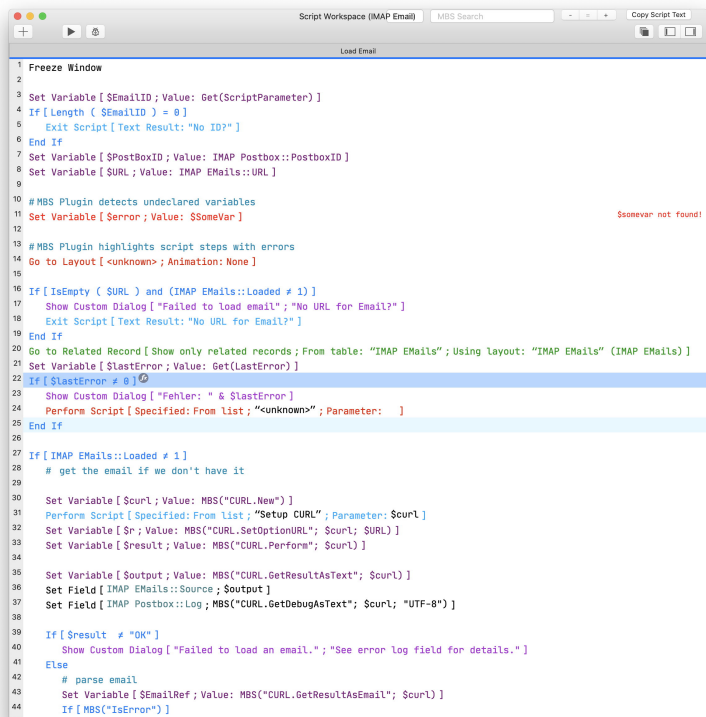
```
1 Freeze Window
2
3 Set Variable [ $EmailID ; Value: Get( ScriptParameter ) ]
4 If [ Length ( $EmailID ) = 0 ]
5     Exit Script [ Text Result: "No ID?" ]
6 End If
7 Set Variable [ $PostboxID ; Value: IMAP Postbox::PostboxID ]
8 Set Variable [ $URL ; Value: IMAP Emails::URL ]
9
10 #MBS Plugin detects undeclared variables
11 Set Variable [ $error ; Value: $SomeVar ]
12
13 #MBS Plugin highlights script steps with errors
14 Go to Layout [ <unknown> ; Animation: None ]
15
16 If [ IsEmpty ( $URL ) and ( IMAP Emails::Loaded ≠ 1 ) ]
17     Show Custom Dialog [ "Failed to load email"; "No URL for Email?" ]
18     Exit Script [ Text Result: "No URL for Email?" ]
19 End If
20 Go to Related Record [ Show only related records ; From table: "IMAP Emails" ; Using layout: "IMAP Emails" (IMAP Emails) ]
21 Set Variable [ $LastError ; Value: Get( LastError ) ]
22 If [ $LastError ≠ 0 ]
23     Show Custom Dialog [ "Fehler: " & $LastError ]
24     Perform Script [ Specified: From List ; "unknown" ; Parameter: ]
25 End If
26
27 If [ IMAP Emails::Loaded ≠ 1 ]
28     # get the email if we don't have it
29
30     Set Variable [ $curl ; Value: MBS("CURL.New") ]
31     Perform Script [ Specified: From List ; "Setup CURL" ; Parameter: $curl ]
32     Set Variable [ $r ; Value: MBS("CURL.SetOptionURL"; $curl; $URL) ]
33     Set Variable [ $result ; Value: MBS("CURL.Perform"; $curl) ]
34
35     Set Variable [ $output ; Value: MBS("CURL.GetResultAsText"; $curl) ]
36     Set Field [ IMAP Emails::Source ; $output ]
37     Set Field [ IMAP Postbox::Log ; MBS("CURL.GetDebugAsText"; $curl; "UTF-8") ]
38
39     If [ $result ≠ "OK" ]
40         Show Custom Dialog [ "Failed to load an email."; "See error log field for details." ]
41     Else
42         # parse email
43         Set Variable [ $EmailRef ; Value: MBS("CURL.GetResultAsEmail"; $curl) ]
44         If [ MBS("IsError") ]
45             Show Custom Dialog [ "Failed to get an email."; "See error log field for details." ]
46         End If
```

On the left is pure FileMaker and on the right side is with [MBS FileMaker Plugin](#). Below in light mode:



The screenshot shows the FileMaker Script Workspace in light mode with the MBS FileMaker Plugin installed. The script is titled 'Load Email' and contains 51 lines of code. The code is syntax-highlighted, with different colors used for keywords, strings, and comments. The interface includes a toolbar at the top and a status bar at the bottom.

```
1 Freeze Window
2
3 Set Variable [ $EmailID ; Value: Get( ScriptParameter ) ]
4 If [ Length ( $EmailID ) = 0 ]
5     Exit Script [ Text Result: "No ID?" ]
6 End If
7 Set Variable [ $PostboxID ; Value: IMAP Postbox::PostboxID ]
8 Set Variable [ $URL ; Value: IMAP Emails::URL ]
9
10 #MBS Plugin detects undeclared variables
11 Set Variable [ $error ; Value: $SomeVar ]
12
13 #MBS Plugin highlights script steps with errors
14 Go to Layout [ <unknown> ; Animation: None ]
15
16 If [ IsEmpty ( $URL ) and ( IMAP Emails::Loaded ≠ 1 ) ]
17     Show Custom Dialog [ "Failed to load email"; "No URL for Email?" ]
18     Exit Script [ Text Result: "No URL for Email?" ]
19 End If
20 Go to Related Record [ Show only related records ; From table: "IMAP Emails" ; Using layout: "IMAP Emails" (IMAP Emails) ]
21 Set Variable [ $LastError ; Value: Get( LastError ) ]
22 If [ $LastError ≠ 0 ]
23     Show Custom Dialog [ "Fehler: " & $LastError ]
24     Perform Script [ Specified: From List ; "unknown" ; Parameter: ]
25 End If
26
27 If [ IMAP Emails::Loaded ≠ 1 ]
28     # get the email if we don't have it
29
30     Set Variable [ $curl ; Value: <function Missing>("CURL.New") ]
31     Perform Script [ Specified: From List ; "Setup CURL" ; Parameter: $curl ]
32     Set Variable [ $r ; Value: <function Missing>("CURL.SetOptionURL"; $curl; $URL) ]
33     Set Variable [ $result ; Value: <function Missing>("CURL.Perform"; $curl) ]
34
35     Set Variable [ $output ; Value: <function Missing>("CURL.GetResultAsText"; $curl) ]
36     Set Field [ IMAP Emails::Source ; $output ]
37     Set Field [ IMAP Postbox::Log ; <function Missing>("CURL.GetDebugAsText"; $curl; "UTF-8") ]
38
39     If [ $result ≠ "OK" ]
40         Show Custom Dialog [ "Failed to load an email."; "See error log field for details." ]
41     Else
42         # parse email
43         Set Variable [ $EmailRef ; Value: <function Missing>("CURL.GetResultAsEmail"; $curl) ]
44         If [ <function Missing>("IsError") ]
45             Show Custom Dialog [ "Failed to get an email."; "See error log field for details." ]
46         Else
47             Perform Script [ Specified: From List ; "ParseEmail"; Parameter: $EmailRef ]
48         End If
49     End If
50     Set Variable [ $r ; Value: <function Missing>("CURL.Cleanup"; $curl) ]
51 End If
```



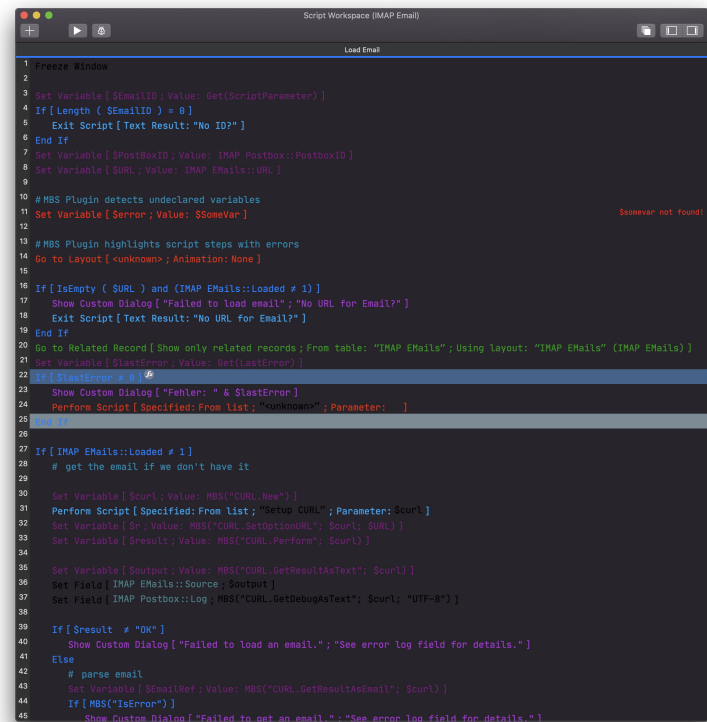
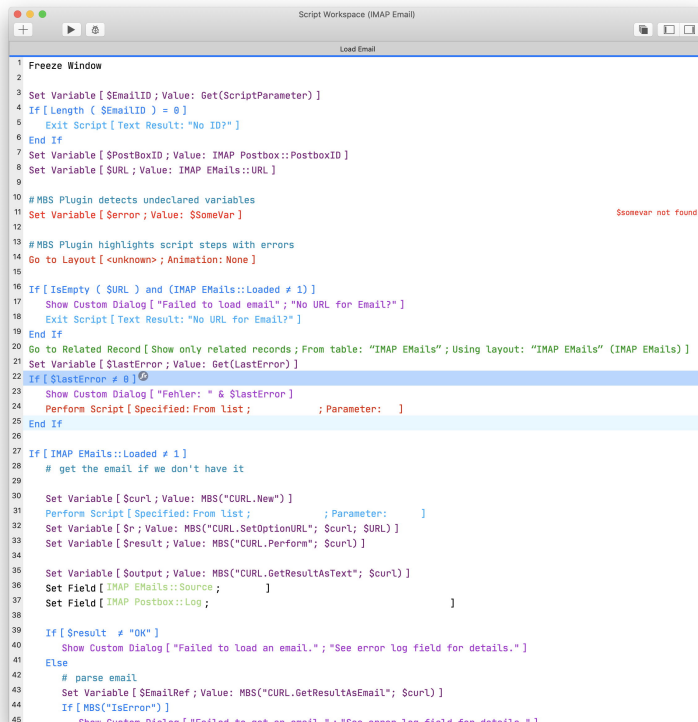
The screenshot shows the FileMaker Script Workspace in light mode with the MBS FileMaker Plugin installed. The script is titled 'Load Email' and contains 46 lines of code. The code is syntax-highlighted, with different colors used for keywords, strings, and comments. The interface includes a toolbar at the top with a search bar and a status bar at the bottom. A red error message 'Somevar not found!' is visible on the right side of the script.

```
1 Freeze Window
2
3 Set Variable [ $EmailID ; Value: Get( ScriptParameter ) ]
4 If [ Length ( $EmailID ) = 0 ]
5     Exit Script [ Text Result: "No ID?" ]
6 End If
7 Set Variable [ $PostboxID ; Value: IMAP Postbox::PostboxID ]
8 Set Variable [ $URL ; Value: IMAP Emails::URL ]
9
10 #MBS Plugin detects undeclared variables
11 Set Variable [ $error ; Value: $SomeVar ]
12
13 #MBS Plugin highlights script steps with errors
14 Go to Layout [ <unknown> ; Animation: None ]
15
16 If [ IsEmpty ( $URL ) and ( IMAP Emails::Loaded ≠ 1 ) ]
17     Show Custom Dialog [ "Failed to load email"; "No URL for Email?" ]
18     Exit Script [ Text Result: "No URL for Email?" ]
19 End If
20 Go to Related Record [ Show only related records ; From table: "IMAP Emails" ; Using layout: "IMAP Emails" (IMAP Emails) ]
21 Set Variable [ $LastError ; Value: Get( LastError ) ]
22 If [ $LastError ≠ 0 ]
23     Show Custom Dialog [ "Fehler: " & $LastError ]
24     Perform Script [ Specified: From List ; "unknown" ; Parameter: ]
25 End If
26
27 If [ IMAP Emails::Loaded ≠ 1 ]
28     # get the email if we don't have it
29
30     Set Variable [ $curl ; Value: MBS("CURL.New") ]
31     Perform Script [ Specified: From List ; "Setup CURL" ; Parameter: $curl ]
32     Set Variable [ $r ; Value: MBS("CURL.SetOptionURL"; $curl; $URL) ]
33     Set Variable [ $result ; Value: MBS("CURL.Perform"; $curl) ]
34
35     Set Variable [ $output ; Value: MBS("CURL.GetResultAsText"; $curl) ]
36     Set Field [ IMAP Emails::Source ; $output ]
37     Set Field [ IMAP Postbox::Log ; MBS("CURL.GetDebugAsText"; $curl; "UTF-8") ]
38
39     If [ $result ≠ "OK" ]
40         Show Custom Dialog [ "Failed to load an email."; "See error log field for details." ]
41     Else
42         # parse email
43         Set Variable [ $EmailRef ; Value: MBS("CURL.GetResultAsEmail"; $curl) ]
44         If [ MBS("IsError") ]
45             Show Custom Dialog [ "Failed to get an email."; "See error log field for details." ]
46         End If
```

You may notice that our plugin applies coloring rules, e.g. to highlight broken script lines in red. The plugin colors comments based on tags, highlights blocks and shows errors for undefined variables being used.

Learn more about [Syntax Coloring](#)

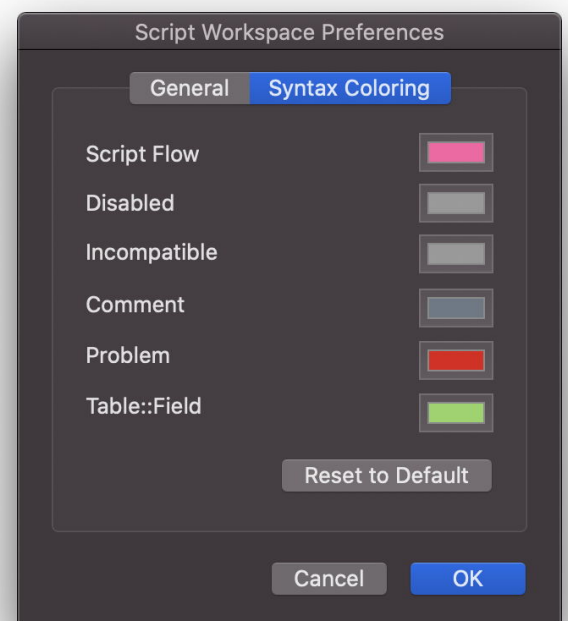
If you run in the screens below, the colors are not right and you should reset them:



In FileMaker Pro, please go to the Script Workspace Preferences dialog via view menu and press the reset button. You may need to do this whenever you switch between dark and light mode.

The [MBS FileMaker Plugin](#) rules can be reset at anytime with running MBS("[SyntaxColoring.FactoryDefaults](#)") function via data viewer or script. This installs the default rule sets for light and dark mode. If you have custom color rules, you may need to install them again. And if you use [fmSyntaxColorizer](#) or another database to manage rules for syntax colorization, please go there to apply your rules again.

The syntax coloring features are provided by [MBS FileMaker Plugin](#) for



free on macOS to all FileMaker users installing the plugin. We still appreciate if you buy a license and use all the other 6000 functions. They can be configured via [preferences dialog](#) or our [SyntaxColoring](#) function set.

For more details on over 20 improvements to FileMaker IDE on macOS, please check the [Syntax Coloring](#) website.

Please do not hesitate to contact us with your questions.