

MBS FileMaker Plugin

Christian Schmitz
Monkeybread Software

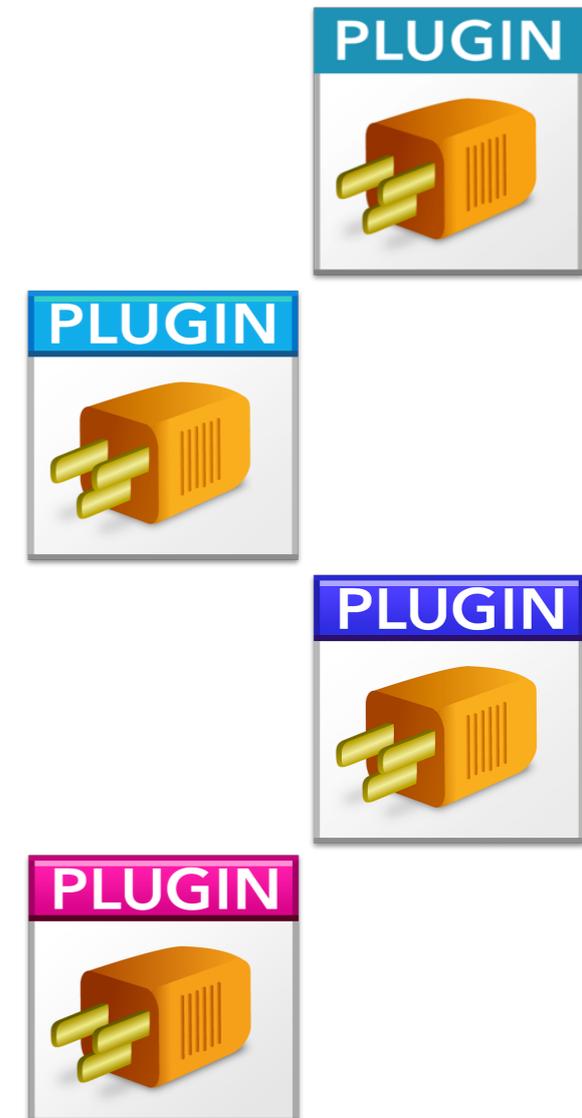
Monkeybread Software

- MBS FileMaker Plugin
- Seit 2006
 - **19 Jahre**
- Über 7700 Funktionen
 - 200 neu in 2025
 - 300 neu in 2024
- Über 600 Beispieldatenbanken



MBS FileMaker Plugin

- Eine Plugindatei
 - Nur benutzte Teile werden geladen
- Einfach zu installieren und aktualisieren
 - Plugin-Datei installieren [MBS::PluginMac]
- Für viele FileMaker Versionen
 - FileMaker 7 bis zur aktuellen Version 22.0
 - FileMaker 22.0 unterstützt seit Version 15.2



MBS FileMaker Plugin

- FileMaker Pro
- FileMaker Server
 - Script auf Server ausführen
 - Scheduled Scripts
 - Data API
 - WebDirect
 - MacOS, Windows & Linux
- FileMaker iOS SDK
 - iPhone, iPad, MacOS & Vision Pro
- FileMaker Go nur über Script auf Server ausführen
- FMDataMigration, FMDeveloperTool und FMUpgradeTool



MBS FileMaker Plugin

- Eine zentrale MBS Funktion und ein MBS Scriptschritt
 - Erster Parameter = Funktionsname
 - Kann berechnet werden oder aus Feld kommen
- Auf Fehler prüfen mit `MBS("IsError")`
 - Fehlermeldungen mit [MBS] Präfix
 - Fehler aufzeichnen mit `ClearErrors`, `HadErrors` & `ErrorLog`
- Referenznummer
 - z.B. `JSON.Parse`, `JSON.Format` & `JSON.Release`

Wartung

- Regelmäßige für neue FileMaker-Versionen aktualisiert
 - Vorbereitungen für die nächste Version
- ARM-Version für Linux
 - Ein Linux-Plugin für alle Ubuntu-Versionen.
- Regelmäßige Aktualisierung der Bibliotheken
 - DynaPDF, OpenSSL, CURL, SSH, Archive, SQLAPI, SQLite und andere

Custom Functions

- Registrieren im Start Script
- Verwendbar in allen Dateien
- FileMaker Pro, Server und iOS SDK
- Variable Anzahl Parameter möglich
- Implementierung in FileMaker oder JavaScript

```
▼ MBS Plug-In
  Concat(Value1; Value2)
  ConcatJS(Value1; Value2)
  ConcatWK(Value1; Value2)
  Join(Value; Value)
  JSONTest1(a; b)
  MBS(FunctionName)
  SplitJoin(InputText; Delimiter1; Deli
  Square(value)
  TestCounter()
  TestCounterWK()
```

FileMaker Custom Functions

Records: 1 of 10 Total (Unsorted) | Show All | New Record | Delete Record | Find | Sort | Share | Search

Layout: CustomFunctions | View As: [Grid] [Table] [Form] | Preview | Edit Layout

FileMaker Custom Functions via MBS Plugin

We can store custom functions in a table and load them into memory and call them from everywhere. Multiple parameters possible and data types are preserved.

Name	Concat	FunctionID	3
Declaration	Concat(Value1; Value2)	MinParameters	2
Description	A quick test function to concat two texts	MaxParameters	2
Parameters	Value1	The first parameter	Hello
	Value2	The second parameter	World
	Name	Description	Value
Body	Value1 & Value2		
Result			
Type	FileMaker	Format	Check Register Function Register and Try
SampleCall	Concat(Value1; Value2)		

FileMaker Custom Functions

Records: 4 / 10 Total (Unsorted)

Layout: CustomFunctions | View As: [Grid] [Table] [Preview] | Search

FileMaker Custom Functions via MBS Plugin

We can store custom functions in a table and load them into memory and call them from everywhere. Multiple parameters possible and data types are preserved.

Name	Join	FunctionID	6
Declaration	Join(Value; Value)	MinParameters	0
Description	Join values to one text	MaxParameters	-1

Parameters

Value	Description	Hello
Value	Description	World
Name	Description	Value

Body

```
Let(
  [
    // we need a loop due to the variable number of arguments
    count = MBS( "Plugin.CustomFunctionParameterCount" );
    list = While(
      [ list = ""; index = 0 ];
      index < count;
      [
        list = list & MBS( "Plugin.CustomFunctionParameter"; index );
        index = index + 1
      ];
      list
    );
    list
  ]
);
```

Result

Type: FileMaker | Format | Check | Register Function | Register and Try

SampleCall: Join(Value; Value)

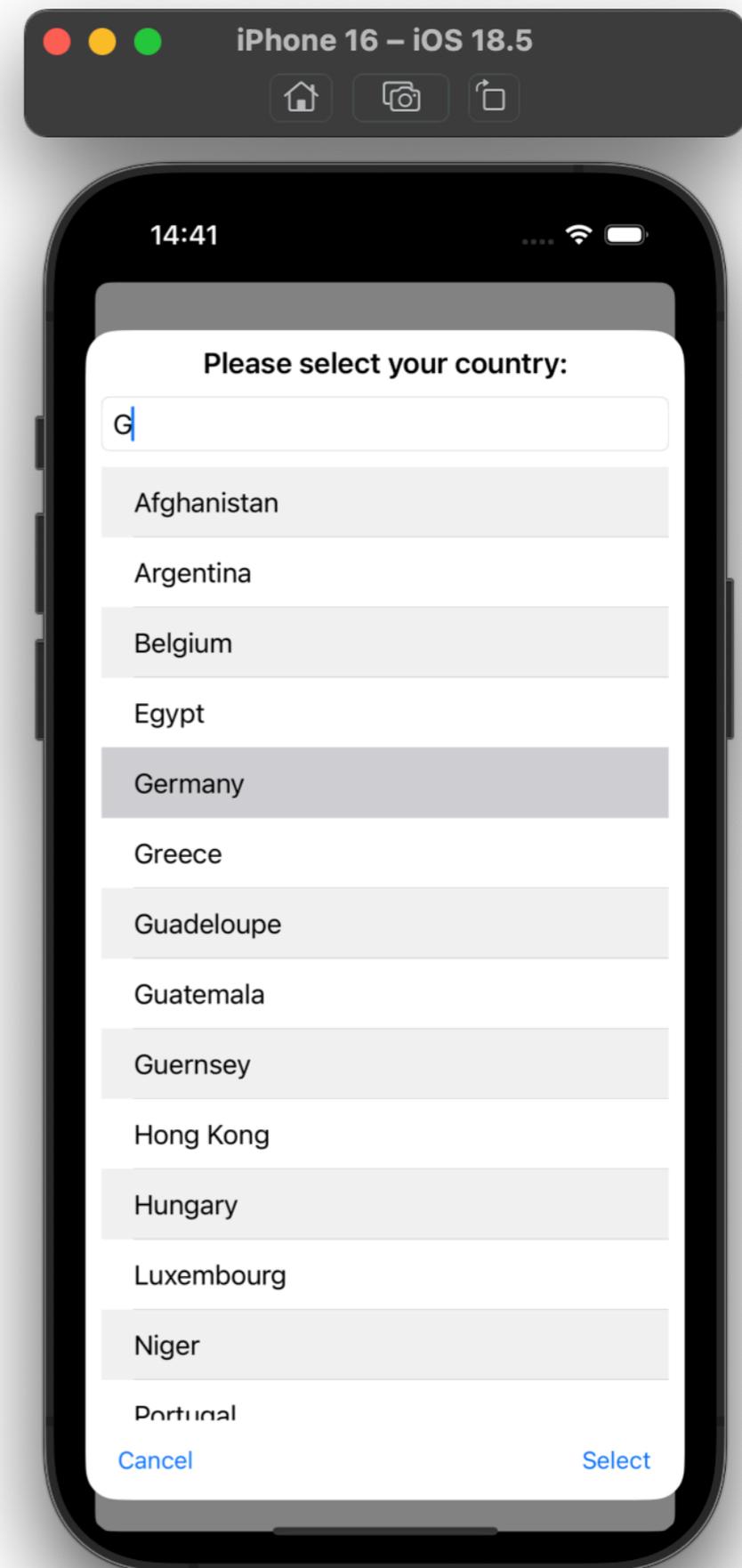
FileMaker Custom Functions via MBS Plugin

We can store custom functions in a table and load them into memory and call them from everywhere. Multiple parameters possible and data types are preserved.

Name	<input type="text" value="ConcatJS"/>	FunctionID	<input type="text" value="7"/>
Declaration	<input type="text" value="ConcatJS(Value1; Value2)"/>	MinParameters	<input type="text" value="2"/>
Description	<input type="text" value="A quick test function to concat two texts in JavaScript"/>	MaxParameters	<input type="text" value="2"/>
Parameters	<input type="text" value="Value1"/>	<input type="text" value="First text"/>	<input type="text" value="Hello"/>
	<input type="text" value="Value2"/>	<input type="text" value="Second text"/>	<input type="text" value="World"/>
	<input type="text" value="Name"/>	<input type="text" value="Description"/>	<input type="text" value="Value"/>
Body	<pre>function ConcatJS(a, b) { return String(a) + String(b); }</pre>		
Result	<input type="text"/>		
Type	<input type="text" value="JavaScript"/>	Register Function	Register and Try
SampleCall	<input type="text" value="ConcatJS(Value1; Value2)"/>		

iOS

- ListDialog für iOS
- Eine Spalte
- Zweite Zeile für Details
- Mehrfachauswahl
- Eigene Buttons
- Filterfeld



Using ChatGPT in FileMaker

The screenshot shows a FileMaker interface titled "ChatGPT". At the top, there is a navigation bar with a "Records" section showing "1 Total (Unsorted)", and buttons for "Show All", "New Record", "Delete Record", "Find", "Sort", and "Share". A search bar is also present. Below the navigation bar, there is a "Layout: ChatGPT" dropdown and "View As" options (table, list, grid) with a "Preview" button. The main content area is divided into four sections:

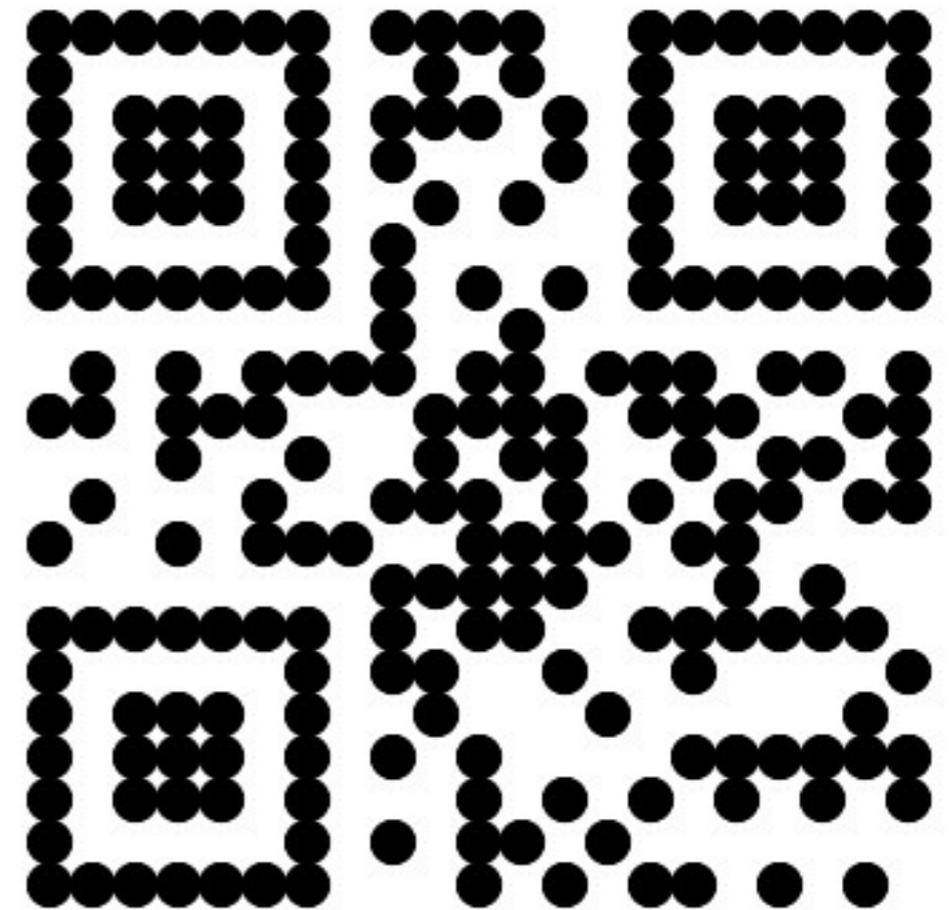
- Query System:** A text input field containing "Please translate text to English."
- Query User:** A text input field containing "Die nächste FileMaker Konferenz in Deutschland findet in Hamburg statt vom 15. bis 17. September 2025." and a "Run Query" button.
- Result:** An empty text area for the query output.
- Debug Log:** An empty text area for debugging information.

At the bottom, there is a form for authentication:

- OpenAI Organization:** A text input field containing "org-xxx".
- Authorization:** An empty text input field.
- Instructions:** "Please put in your organization and bearer token for authentication."

Barcodes

- Barcodes erzeugen und erkennen
- Barcode.DetectAsJSON
- JSON Ausgabe für Vektordaten
 - Beispiel für QR-Code mit Kreisen
- Stacked Barcodes



Saxon

- Use Saxon XML Library in FileMaker
 - XSLT 3.0 für Transformationen
 - XQuery 3.1 für Abfragen
 - XSD 1.1 zum Validieren
 - XPath 3.1 zum Navigieren in Dokumenten
- Saxon Version 12.8
 - Download von MBS Seite
- Optional JSON als Eingabe
- Separate Lizenz, ausprobieren ohne Lizenz

e-Rechnung anzeigen mit Saxon

The screenshot shows the Saxon Visualize electronic Invoices application. The interface includes a top navigation bar with a search bar and various action buttons. Below the navigation bar, there is a main content area with a blue header containing text about using Saxon and the XRechnung Visualization project. The main content area is divided into two columns: Invoice XML and Intermediate XML. The Invoice XML column contains XML code for a ZUGFeRD invoice. The Intermediate XML column contains XML code for an HTML invoice. A 'Show Invoice' button is located between the two columns. Below the XML code, there is a navigation bar with tabs for 'Übersicht', 'Details', 'Zusätze', 'Anlagen', and 'Laufzettel'. The 'Übersicht' tab is currently selected. Below the navigation bar, there is a warning message: 'Wir übernehmen keine Haftung für die Richtigkeit der Daten.' Below the warning message, there is a section titled 'Informationen zum Käufer' with a table of buyer information. The table has columns for the field name and the value. The fields and values are: Käuferreferenz (empty), Name (Kunden AG Mitte), Adresszeile 1 (Kundenstraße 15), Adresszeile 2 (empty), Adresszeile 3 (empty), and PLZ (empty). At the bottom of the application, there is a footer with the XSLFolder path, a download link, and a website link.

Saxon Visualize electronic Invoices

Records: 1 of 2 Total (Unsorted)

Layout: Saxon Vis...Invoices | View As: [Grid] [Table] [Preview] | Edit Layout

We can use Saxon and the XRechnung Visualization project to show invoices as HTML.
UBL, Cross Industry Invoices (ZUGFeRD and Factur-X)

Invoice XML: `<?xml version='1.0' encoding='UTF-8' ?>
<!-- English disclaimer below.-->
<!--Nutzungsrechte
ZUGFeRD Datenformat Version 2.3.0,
18.09.2024
Beispiel Version 18.09.2024`

Intermediate XML: `<?xml version="1.0" encoding="UTF-8"?>`

HTML: `<!DOCTYPE HTML>
<html lang="de">
<head>`

[Show Invoice](#)

Übersicht Details Zusätze Anlagen Laufzettel

Wir übernehmen keine Haftung für die Richtigkeit der Daten.

Informationen zum Käufer

| | |
|-----------------|-----------------|
| Käuferreferenz: | |
| Name: | Kunden AG Mitte |
| Adresszeile 1: | Kundenstraße 15 |
| Adresszeile 2: | |
| Adresszeile 3: | |
| PLZ: | |

XSLFolder: /Users/cs/ZUGFeRD to HTML/xrechnung | Download xrechnung-visualization and point to XSL folder there. [Website](#)

e-Rechnung validieren mit Saxon

ZUGFeRD Validation

Records: 1 Total (Unsorted)
Show All
New Record
Delete Record
Find
Sort
Share

Layout: ZUGFeRD Validation
View As: [Grid]
Preview
Edit Layout

We can use Saxon and the XSL for ZUGFeRD to validate the structure and the business rules. Structure would complain about a missing or duplicate line. But the business rules also looks if values are plausible.

Invoice XML

```
<?xml version='1.0' encoding='UTF-8' ?>
<!-- English disclaimer below.-->
<!--Nutzungsrechte
ZUGFeRD Datenformat Version 2.3.0, 18.09.2024
Beispiel Version 18.09.2024

Zweck des Forums elektronisch Rechnung Deutschland,
welches am 31. März 2010 unter der Arbeitsgemeinschaft für
wirtschaftliche Verwaltung e. V. gegründet wurde, ist u. a. die
Schaffung und Spezifizierung eines offenen Datenformats
für strukturierten elektronischen Datenaustausch auf der
Grundlage offener und nicht diskriminierender, standardisierter
Technologien ("ZUGFeRD Datenformat")
```

Validate XML Structure with LibXML

Validate XML Structure with Saxon

Validate XML Content

Validation Report - 1 Errors

In content of element <SellerTradeParty>: The content model does not allow element <ram:Name> to appear more than once.

/Q{urn:un:unece:uncefact:data:standard:CrossIndustryInvoice:100}CrossIndustryInvoice[1]/Q{urn:un:unece:uncefact:data:standard:CrossIndustryInvoice:100}SupplyChainTradeTransaction[1]/Q{urn:un:unece:uncefact:data:standard:ReusableAggregateBusinessInformationEntity:100}ApplicableHeaderTradeAgreement[1]/Q{urn:un:unece:uncefact:data:standard:ReusableAggregateBusinessInformationEntity:100}SellerTradeParty[1]/Q{urn:un:unece:uncefact:data:standard:ReusableAggregateBusinessInformationEntity:100}Name[2]

Result

[MBS] XQDY0027: One validation error was reported: In content of element <SellerTradeParty>: The content model does not allow element <ram:Name> to appear more than once. .

Report

HTML

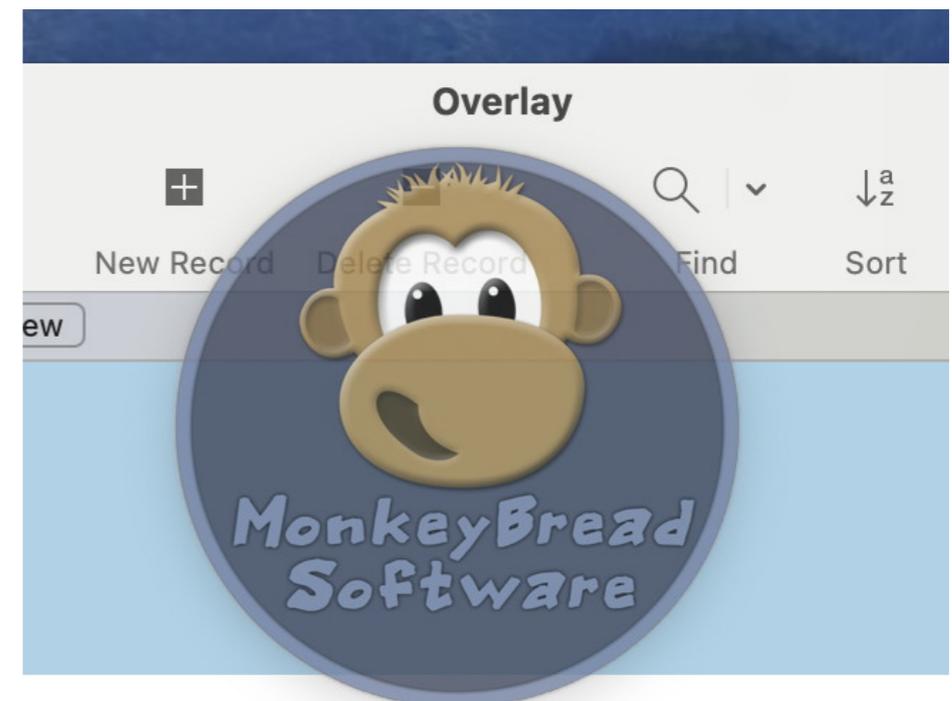
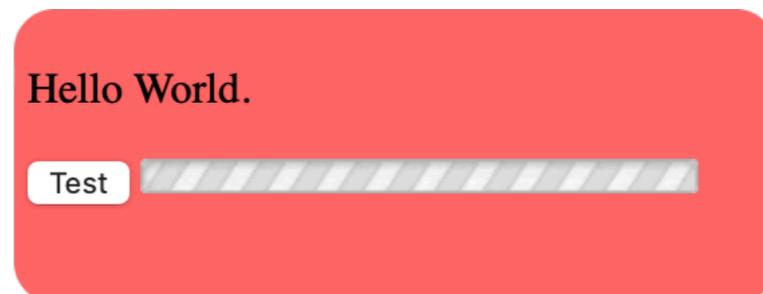
HTML XSLT1

HTML XSLT2

Folder

Overlay windows

- Benutzerdefinierte Fenster ohne Steuerelemente und Rahmen
- Transparenz
- Maus-Betreten/Verlassen und Klick Script-Trigger
- Positionierung über Steuerelemente oder Koordinaten
- Splash Screens, Tutorials, Navigation, Führungslinien oder kontextbezogene Hilfe
- Drag & Drop-Unterstützung
- WebViewer

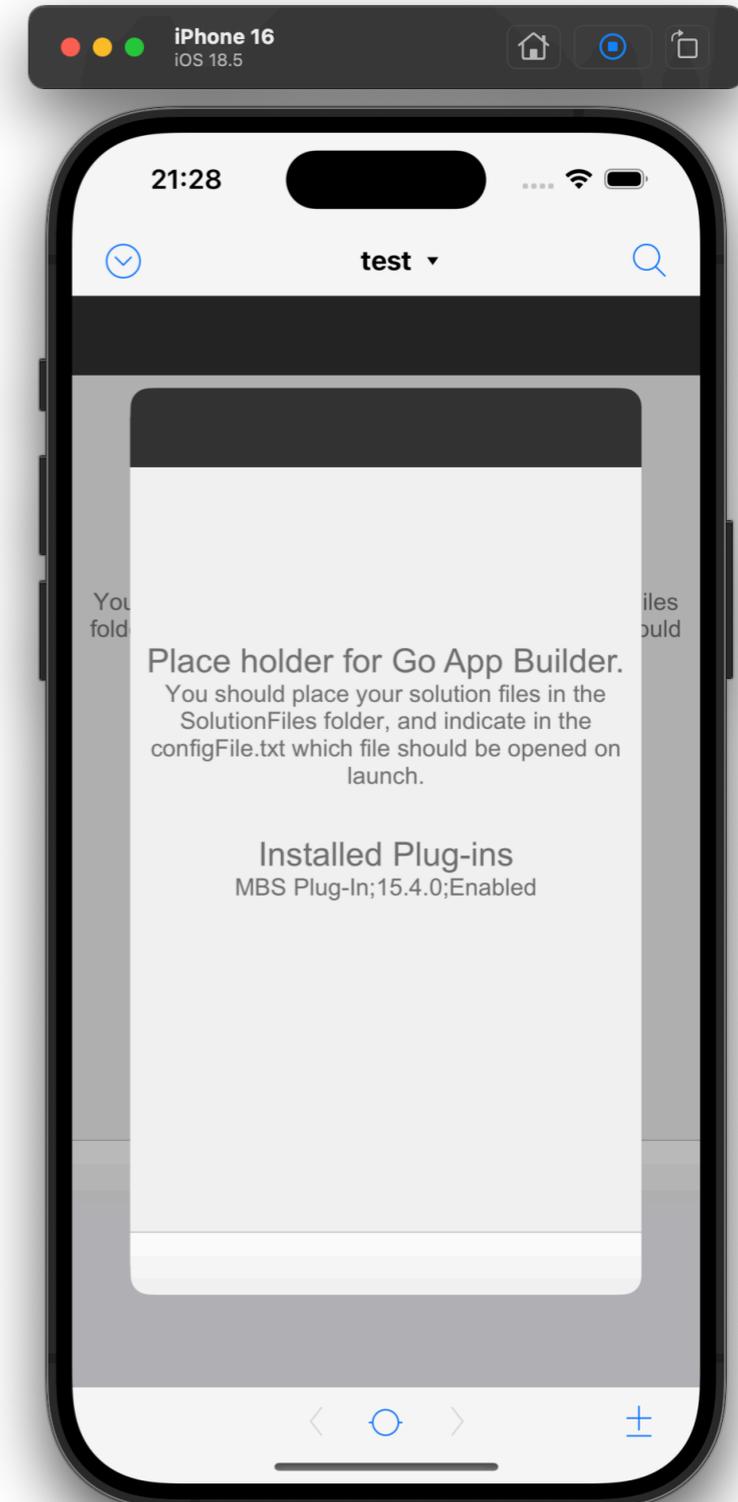
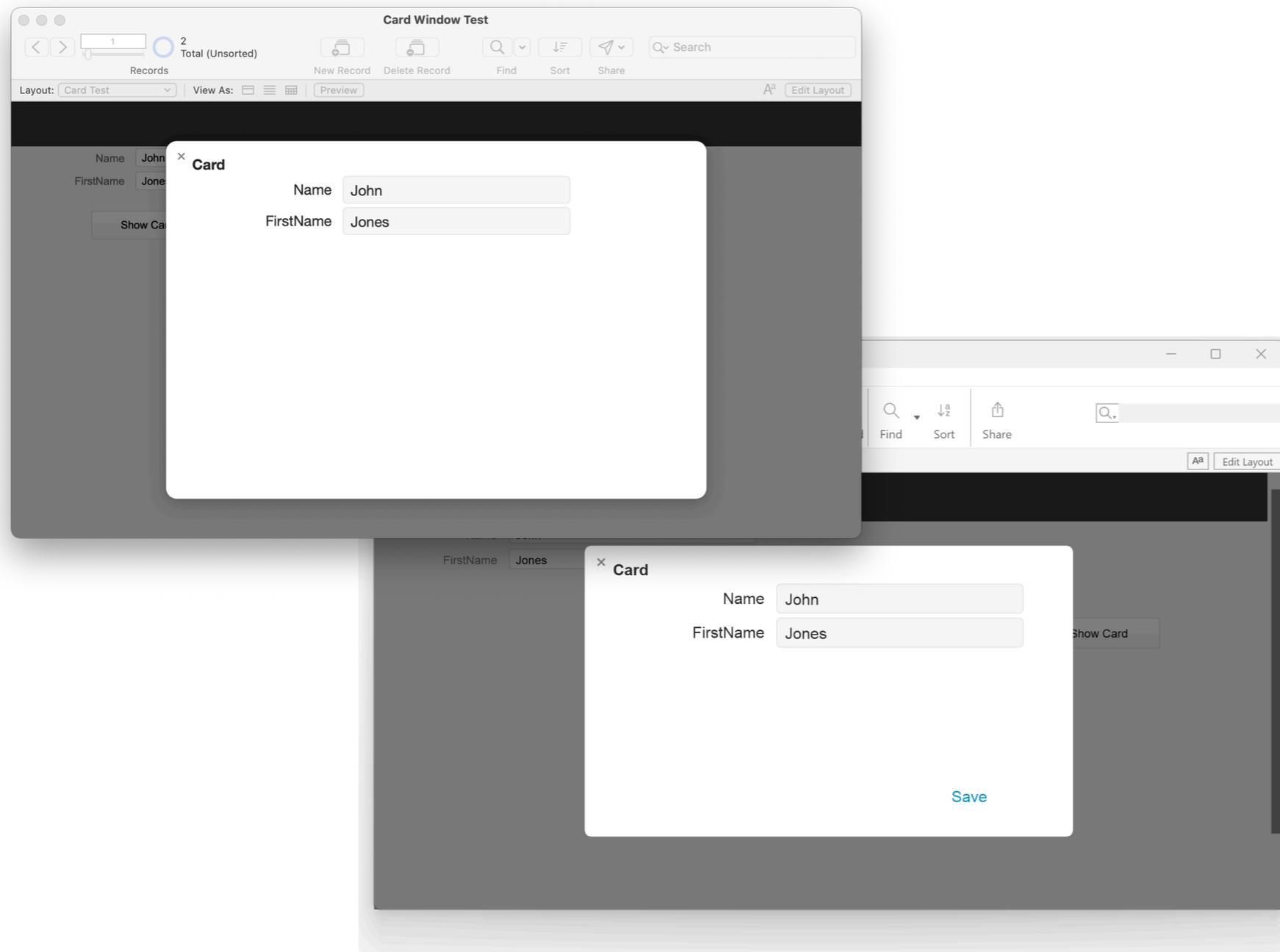


DynaPDF

- ZUGFeRD Beispiele
- Complex Rendering für Windows
 - Für Rechts nach Links Sprachen wie Hebräisch oder Kurdisch
- RenderJob
 - PDF Seiten als Bilder ausgeben mit Hintergrundthread
- Content Parser
 - Text finden und ersetzen
 - Bilder extrahieren, ersetzen oder löschen
- Inkrementelle Updates möglich

Runde Ecken für Karten

- Mac, Windows und iOS SDK



CURL

- Neue Beispiele für Email
 - Gmail mit oAuth
 - Office 365 mit oAuth
- IMAP Upload Flags
- SecureTransport veraltet für macOS

JSON

- Insert, Update oder Delete per JSON
 - **MBS**("JSON.InsertRecord"; "";
"Contacts"; "{\\"FirstName\\": \\"John\\", \\"TheNumber\\": 123}")
 - Einen oder mehrere Datensätze
- JSON.ArraySlice
 - Array teilweise auslesen in neues Array
- JSON.ToCSV
- JSON.SetVariables und JSON.ClearVariables

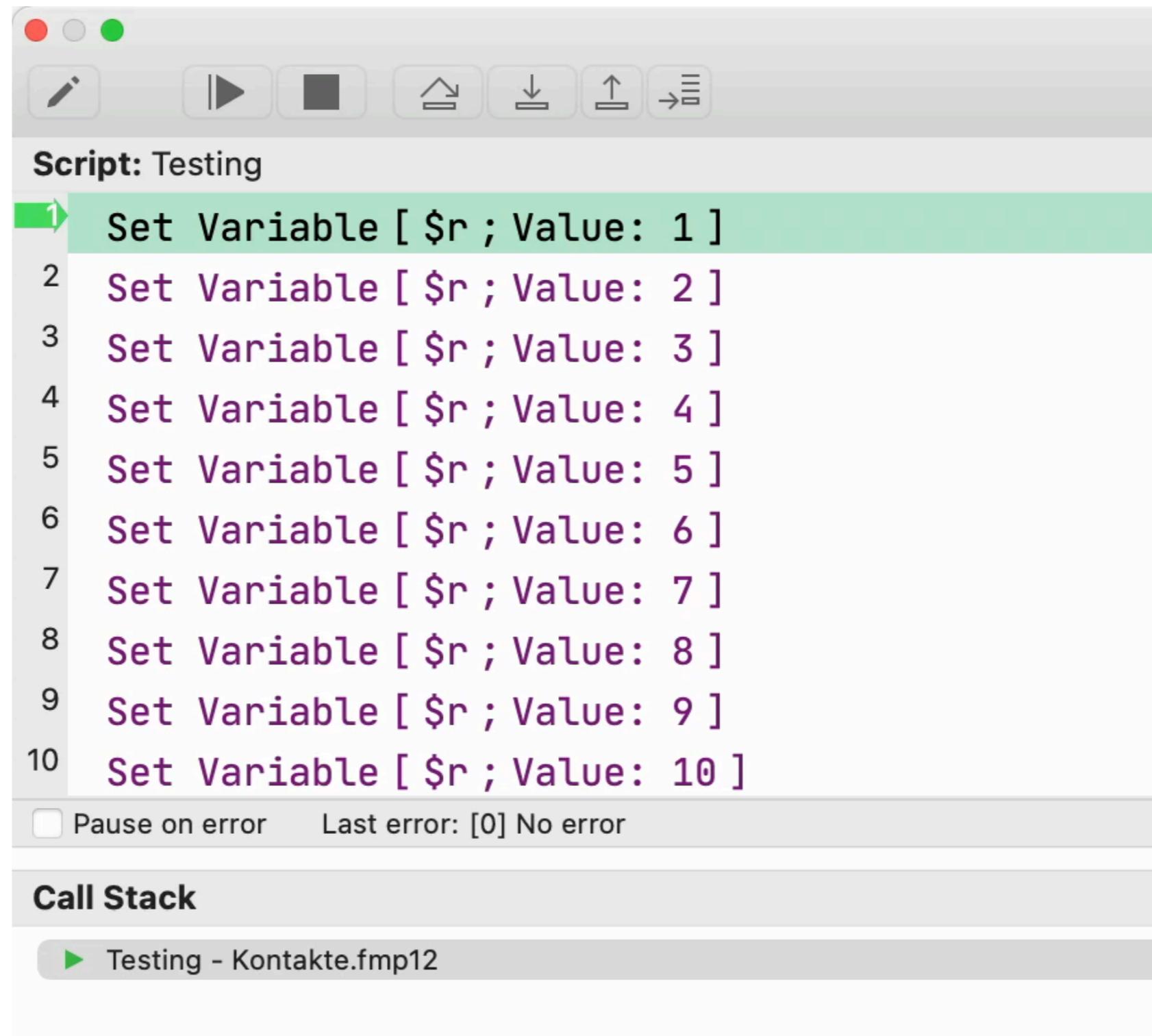
Kleinere Sachen

- Tidy für HTML und XML
 - Text.HTMLtoJSON
- Text.FilterSwissQRCode
- Vision.RectifyDocument
- Und viele mehr

FileMaker ScriptWorkspace- Erweiterungen

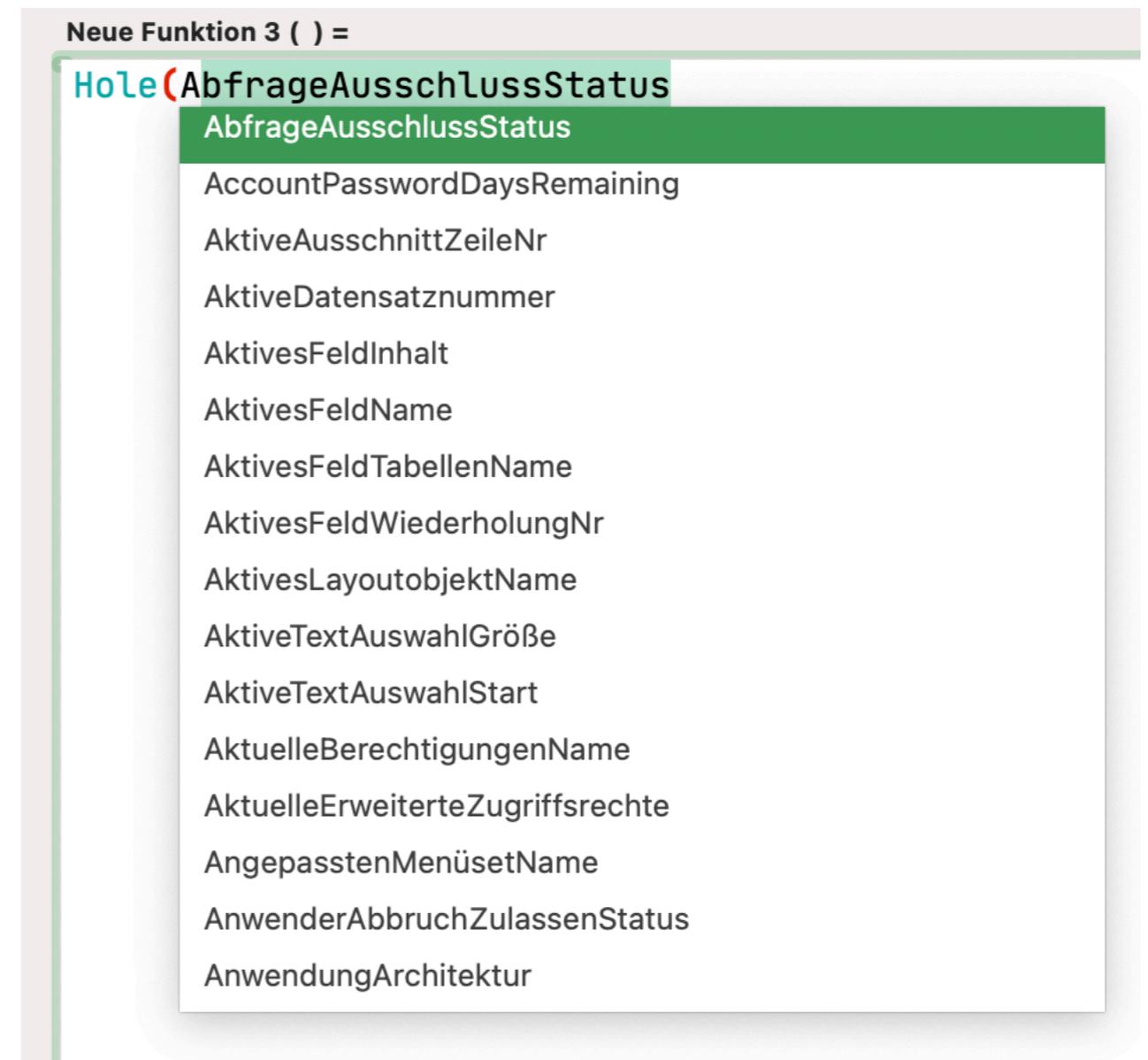
- Über 40 Verbesserungen
 - monkeybreadsoftware.com/filemaker/SyntaxColoring/
- Neue Dinge:
 - Clipboard Converter (Mac & Windows)
 - Kommentarlinks mit Suche: search:
 - Control-Cursor Tasten im Texteditor.
 - Kommentare für Tabellen
 - Tastaturkürzel für Ergebnistyp für Formelfelder
 - Feldauswahl Dialoge mit Abkürzung

Automatisches Scrollen im Debugger

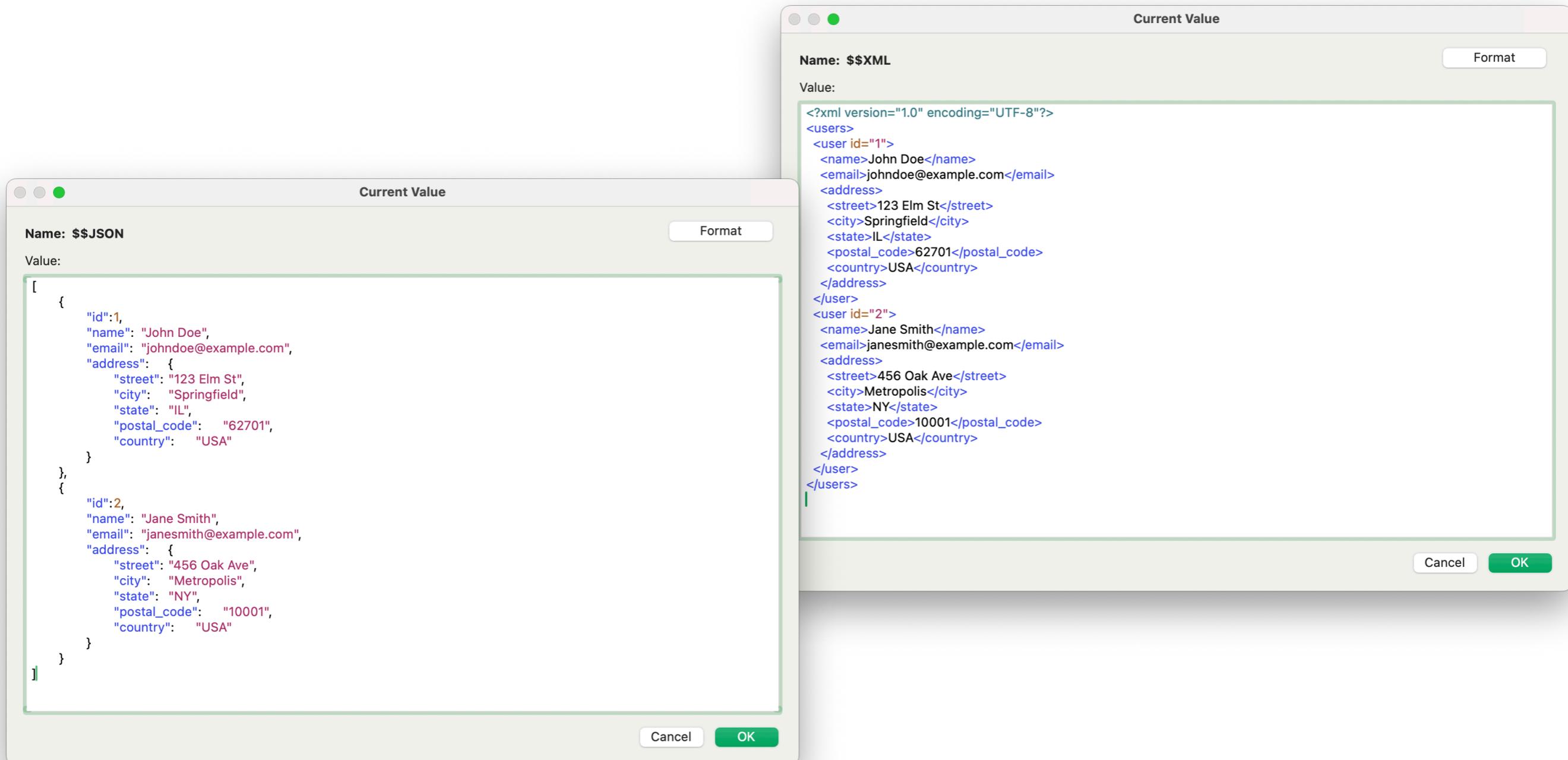


Autovervollständigung für Eigene Funktionen

- Tabellen und Feldnamen
- FileMaker Funktionsnamen
- MBS Funktionen
 - Und Parameter
- Hole() Funktion
- Operatoren



XML und JSON formatieren im Data Viewer



Ressourcen

- Webseite
monkeybreadsoftware.com/filemaker/
- Dokumentation
mbsplugins.eu
- Blog
mbsplugins.de/filemaker
- Videos
monkeybreadsoftware.com/filemaker/video
- Anmeldung für Newsletter oder Testlizenzen auf der Webseite

Konferenzangebot

- 20% Rabatt für neue Lizenzen mit Coupon Code **FileMakerKonferenz**
- MBS Plugin und DynaPDF Lizenzen

FMTraining.tv

- Über 70 Livestreams mit MBS Plugins!



Vielen Dank unseren Sponsoren und Konferenz-Partnern

