FMK | FileMaker Konferenz

C.R.U.D Implementation in FileMaker

… und warum das wichtig ist.

Philipp A. Puls — 72solutions GmbH

Mag. Philipp A. Puls

Geschäftsführender Gesellschafter der 72solutions GmbH

- geschäftsführender Gesellschafter und Mitbegründer der 72solutions GmbH
- Hintergrund in theoretischer Physik von der Universität Wien und der University of Edinburgh.
- 72solutions GmbH ist Platinum-Partner von Claris FileMaker und ein „SBA“-Partner mit seinen Lösungen «base72 Toolbox» und «onexio».
- FileMaker-Karriere 1992 mit Version 2.1 begonnen und seit 2015 durchgehend zertifiziert

Zertifiziert für:

Warum mir das wichtig ist

… und was ist ‘das’?

… Ich bin faul

- Ich hasse Verschwendung

  Verschwendung von Zeit und Aufwand

  Verschwendung von geistiger Kapazität auf Prozesse, die ich schon optimiert habe

  — einmal

  — irgendwo

- Deswegen versuche ich, Code an so vielen Stellen wie möglich wieder zu verwenden. Dadurch muss ich ihn an so wenig Stellen wie nötig verbessern.

Zum Beispiel

… Suchen ist einfach, oder?

Weil… wie schwierig kann Suchen denn sein?

- Go to Layout [“LayoutName”]
- Enter Find Mode []
- Set Field [“FieldName”; “FieldValue”]
- Perform Find []

… wie schwierig kann Suchen denn sein?

- *Enter Find Mode []*
- Go to Layout [“LayoutName”]
- ~~Enter Find Mode []~~
- Set Field [“FieldName”; “FieldValue”]
- Perform Find []

… wie schwierig kann Suchen denn sein?

- *Set Error Capture [“ON”]*
- Enter Find Mode []
- Go to Layout [“LayoutName”]
- Set Field [“FieldName”; “FieldValue”]
- Perform Find []
- *Set Error Capture [“OFF”]*

… wie schwierig kann Suchen denn sein?

- Set Error Capture [“ON”]
- *New Window [“Document”/“Card/…”*
- Enter Find Mode []
- Go to Layout [“LayoutName”]
- Set Field [“FieldName”; “FieldValue”]
- Perform Find []
- Set Error Capture [“OFF”]

… wie schwierig kann Suchen denn sein?

- Set Error Capture [“ON”]
- New Window [“Document”/“Card/…”
- *Um Fensterposition kümmern*
- *Um das Zoom-Level des Users kümmern*
- Enter Find Mode []
- Go to Layout [“LayoutName”]
- Set Field [“FieldName”; “FieldValue”]
- Perform Find []
- Set Error Capture [“OFF”]

… wie schwierig kann Suchen denn sein?

- Set Error Capture [“ON”]
- New Window [“Document”/“Card/…”
- Um Fensterposition kümmern
- Um das Zoom-Level des Users kümmern
- Enter Find Mode []
- Go to Layout [“LayoutName”]
- Set Field [“FieldName”; “FieldValue”]
- Perform Find []
- Set Error Capture [“OFF”]
- *Go To Layout [“TargetLayoutName”] //List //Detail abhängig von FoundCount?*

… wie schwierig kann Suchen denn sein?

- Set Error Capture [“ON”]
- New Window [“Document”/“Card/…”
- Um Fensterposition kümmern
- Um das Zoom-Level des Users kümmern
- Enter Find Mode []
- Go to Layout [“LayoutName”]
- Set Field [“FieldName”; “FieldValue”]
- Perform Find []
- Set Error Capture [“OFF”]
- *Go To Layout [“TargetLayoutName”] //List //Detail abhängig von FoundCount?*
- *Go To Record [“LandingRecord”] //wenn mehr als einer gefunden wurde, welcher ist aktiv*

Woher kommt unser Ansatz

… das haben andere schon vor uns gelöst.

Inspiration kommt aus dem wahren Leben

- RESTful Calls über HTTP(S) verwenden GET, POST, PUT and DELETE Befehle.

Diese ähneln den Basis-Operationen auf persistenten Speicher

C … create

R … read

U … update

D … delete

- Und in gewisser Weise sind wir alle hier im Geschäft der persistenten Speicherung.

Was wir daraus gemacht haben

Die 72s API Scripts

Inspiration kommt aus dem wahren Leben

- **CreateEdit**

  Editieren eines Datensatzes nach einer Suche. Natürlich kann man auch nicht suchen und einfach einen Neuen erzeugen.

- **FindPresent**

  Suchen und Anzeigen eines Datensatzes in einem Kontext/Layout/Fenster

- **Read**

  Lesen eines Datensatzes mit vorheriger Definition, welche Felder gelesen werden sollen.

- **Delete**

  Finden von Datensätzen und löschen (nach Archivierung), wenn bestimmte Kriterien erfüllt sind.

Notwendiges

Mit welchen Tools das möglich war …

json2var($json ; $namespace)

- **Eigene CustomFunction,** welche …

  jedes JSON in Variablen zerlegt wobei

  jeder JSONkey der Variablen-Name wird

  und jeder JSONvalue der neue Variablen-Wert ist

Zusätzlich wird eine $jsonVarNames gesetzt, eine list_ml aller gerade angelegten Variablen.

Namenskonvention

- Jeder BaseTable hat ein Aufkommen ohne Relationen
- Wir nennen dieses TO «Table»_oB
- Das dazugehörige Layout wird All_«Table» genannt

Ergebnis

- Wir können errechnen, in welches Layout wir wechseln müssen, um schnellen Zugriff auf Daten zu haben.

Und damit man nicht durcheinander kommt …

- Felder, die die gleichen Informationen erhalten, müssen auch gleich heißen.

Konseqenz:

Das bedeutet, dass eine ID in Tabelle A «NrTabelleA» genannt wird, und wenn diese auf der Tabelle B als Fremdschlüssel vorkommt, ändert sie den Namen nicht.

Wie nutze ich das

FindPresent_API

Der Parameter

FindPresent_API: Der Parameter

```
JSONSetElement ( "" ;
["NrDS" ; $NrDS ; JSONnumber ];                    // Die Datensätze IDs (können auch |ed sein)
["Table" ; “base72 File/TargetTable" ; JSONstring ];
["TargetWindow" ; "Main" ; JSONstring ];
[“TargetLayout" ; "2-1" ; JSONstring ];
["SearchOption" ; “suchen/einschränken/ersweitern“ ; JSONstring ];
["PopUpleft" ; $left ; JSONnumber ];
["PopUptop" ; $top ; JSONnumber ];
["PopUpHight" ; $height ; JSONnumber ];
["PopUpWidth" ; $width ; JSONnumber ];
["LandingRecord" ; $NrDSLanding ; JSONnumber ]; // Die ID des DS, auf dem ich stehen will
["WindowMode" ; "search/browse/preview"; JSONstring];
["FollowingScript"; "ScriptName"; JSONstring];      // Ein Script, welches auf die gefundenen DS aufgerufen wird
[“FollowingScriptCondition"; "get(foundcount) =1"; JSONstring] // Evaluate() Bedingungen, um das Script auszuführen
)
```

Delete_API

Der Parameter

Delete_API: Der Parameter

```
JSONSetElement ( "" ;
["NrDS" ; $NrDS ; JSONstring];
["Tabelle" ; $Tabelle ; JSONstring];;
["verbose" ; 1 ; JSONnumber];                    // Soll der Benutzer gewarnt werden
["archiv"; 1; JSONnumber];                       // Soll der DS archiviert werden
["NrParent" ; $NrParent ; JSONString];
["TableParent" ; $TableParent ; JSONString];
["expectedFoundCount" ; $Erwartungswert ; JSONnumber];
                    // nur löschen wenn  get(foundcount) ≤ Erwartungswert
["NrArchiv"; $NrArchiv; JSONnumber]
)
```

CreateEdit_API

Der Parameter

Die Aufgabe

Erstelle ein Script,

welches unabhängig von der Struktur / Datenbank / Lösung ist.

Dieses Script soll genau eine Aufgabe erfüllen

und diese Aufgabe erst durch die Parametrisierung umsetzen.

CreateEdit_API: Der Parameter

```
"Invoice"  & "¶" &                                  // Wo der DS angelegt werden soll
                        // ** entweder **
"Keine"                                             // Wenn nicht gesucht werden muss
                        //** oder **
    JSONSetElement ( "" ;                           // Der Parameter für die Suche
                ["Fieldname" ; $Fieldvalue ; JSONString];
                ["Fieldname2" ; $Fieldvalue2 ; JSONString]
)   & "¶" &                  //** und **
JSONSetElement ( "" ;                                       //Die Feldwerte, die gesetzt/geändert werden
                ["Fieldname" ; $Fieldvalue ; JSONString];
                ["Fieldname2" ; $Fieldvalue2 ; JSONString]
)   & "¶" &
"Invoice_oB::NrInvoice|NrInvoice|||BlockNumber|BlockNumberInvoice"
```

CreateEdit_API

Ein Beispielscript

```
Let ( $r =
JSONSetElement ( "" ;
["Dank" ;
"Vielen Dank für Ihr Interesse!" ;
JSONstring ]
) ; $r )
```

Q & A

Exit script ($r)

Vielen Dank unseren Sponsoren

MonkeyBread Software

Claris

FileMaker Magazin

CamelCase
camelcase.com

24U SOFTWARE

JK SOLUTION
Datenbankentwicklung

NORDEND. SWISS