filemakerhacks

Home   About   Disclaimers   Privacy Policy

FM 12 ExecuteSQL: Dynamic Parameters, part 2

POSTED ON MAY 8, 2012 BY Kevin Frank

This is a quick follow-up to part 1, with a couple more observations about dynamic parameters.

Embedded Apostrophes

Here's one I can't believe I forgot to mention the other day: A major ExecuteSQL headache that dynamic parameters can alleviate is the dreaded "embedded apostrophe" problem. In case you aren't familiar with it, if your text string contains an embedded apostrophe, in standard SQL you must escape it by prepending another apostrophe, for example compare these two "standard" (non-dynamic) queries:

Query	Result
ExecuteSQL ("SELECT city FROM customers WHERE name_last = 'O'Malley' " ; "" ; "")	?

Query	Result
ExecuteSQL ("SELECT city FROM customers WHERE name_last = 'O''Malley' " ; "" ; "")	Sabine

escape internal single quotes or apostrophes by prepending another one

name_first	name_last	address	city	state	zip	sales
Thomas	Utesch	2437 W. Lakeside	Montgomery	IL	60538	$10,459.00
Thomas	O'Malley	13 Whisker Alley	Sabine	TX	77777	$21,782.00
Thomas	Grayham	1656 Union Street	Macon	GA	31209	$443.00
Thomas	Fisher	1720 Peachtree Road	Des Plaines	IL	60016	$1,174.00
Thomas	Adler	531 K St.	Springtown	TX	76082	$1.00

As you might expect, you don't have to worry about this if you instead use a dynamic parameter... just quote the search term the way you would any FileMaker text string (i.e., in double quotes) and go about your business.

Query	Result
ExecuteSQL ("SELECT city FROM customers WHERE name_last = ? " ; "" ; "" ; "O'Malley")	Sabine

"IN" Operators

I've been evangelizing dynamic parameters as if they are always superior, but Stephen Dolenski recently pointed out on Friday Night Chat that when it comes to the "IN" operator, the dynamic approach may end up being more restrictive than the standard one.

As you may know, the IN operator provides a way to avoid long chains of "or" tests. For example, instead of this...

```
ExecuteSQL(
  " SELECT MAX ( sales )
    FROM customers
    WHERE state = 'WA' or state = 'OR' or state = 'ID'  "
  ; "" ; ""
)
```

...if you use the IN operator, the query can be streamlined, thus...

```
ExecuteSQL(
  " SELECT MAX ( sales )
```

SEARCH

Search...

RECENT POSTS

- FM 12 ExecuteSQL: Dynamic Parameters, part 2
- FM 12 ExecuteSQL: Dynamic Parameters, part 1
- FM 12 ExecuteSQL: A Cool Tool
- FM 12 ExecuteSQL, part 2
- FM 12 ExecuteSQL, part 1

RECENT COMMENTS

- Kevin Frank on FM 12 ExecuteSQL: Dynamic Parameters, part 2
- Eran Kendler on FM 12 ExecuteSQL: Dynamic Parameters, part 2
- Kevin Frank on FM 12 ExecuteSQL: Dynamic Parameters, part 1
- Juz on FM 12 ExecuteSQL: Dynamic Parameters, part 1
- Eden Morris on FM 12 ExecuteSQL: Dynamic Parameters, part 1

ARCHIVES

- May 2012
- April 2012
- March 2012
- February 2012
- January 2012
- December 2011
- November 2011
- October 2011
- September 2011
- August 2011
- July 2011
- June 2011
- May 2011
- April 2011
- March 2011
- February 2011
- January 2011
- December 2010
- November 2010

CATEGORIES

- ExecuteSQL
- General
- Level: Advanced
- Level: Any
- Level: Beginner

```
      FROM customers
      WHERE state IN ('WA','OR','ID')  "
      ; "" ; ""
)
```
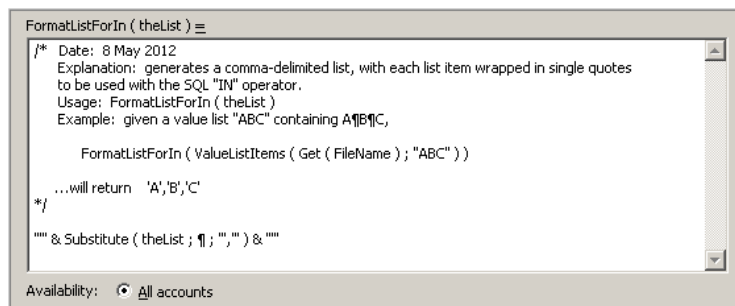
...and with dynamic parameters, it looks like this:

```
ExecuteSQL(
   " SELECT MAX ( sales )
     FROM customers
     WHERE state IN ( ? , ? , ? )  "
     ; "" ; "" ; "WA" ; "OR" ; "ID"
)
```

When you just have a few choices, it's not a big deal to go the dynamic route, but what if you are constructing the IN arguments "on the fly", and you aren't sure how many arguments there will be? I'll come back to that in just a sec, but first I want to introduce a custom function that transforms a return-delimited list into an IN-friendly comma-separated list.

Based on what you saw in part 1, where dynamic parameters could do no wrong, you might think that if you had a value list, Northwest, consisting of WA¶OR¶ID, that this would work...

```
Let (
x = FormatListForIn ( ValueListItems ( Get ( FileName ) ; "Northwest" ) ) ;
ExecuteSQL (
   " SELECT MAX ( sales )
     FROM customers
     WHERE state IN ( ? )  "
     ; "" ; "" ; x )
)  //  end let
```

...but it does not. With dynamic parameters, each item must be enumerated separately — thank you Ralph Lilienkamp for helping me understand this — so you can't use the result of the custom function ('WA','OR','ID') as the corresponding argument for a single "?". Meanwhile, this non-dynamic construction purrs along like a contented kitten:

Or course this is a simple use-case, and your mileage may vary, but I thought it was worth sharing, and I want to close by thanking Stephen Dolenski for bringing it up, and also

Ernest Koe for helping me clarify my thinking about this.

← FM 12 ExecuteSQL: Dynamic Parameters, part 1

2 Responses to *FM 12 ExecuteSQL: Dynamic Parameters, part 2*

Eran Kendler May 10, 2012 at 5:54 pm #

Thank for wonderful info – I'm having fun and tearing hair out using this. Wondering if you know the answer to this :

While above works great for text fields, when trying to using the IN statement on a number field e.g. WHERE ClientID IN … it doesn't seem to work, while it does with text field e.g. WHERE ClientCity IN…

Thanks

(REPLY)

Kevin Frank May 10, 2012 at 6:22 pm #

Hi Eran,

My next blog posting will cover this in more detail… if the underlying field type is numeric, then the CF you're using to assemble the IN string needs to NOT wrap the values in single quotes.

I'll email you a beta of the demo file for my next article and perhaps that will help (but the final one will be better!)

Regards,
Kevin

Leave a Reply

[ ]	Name (required)
[ ]	Email (will not be published) (required)
[ ]	Website

Submit Comment

☐ Notify me of follow-up comments by email.

☐ Notify me of new posts by email.

Powered by WordPress. Designed by WOO THEMES