

Re: Multi Field SQL Where Clause

Source: <http://www.tech-archive.net/Archive/Access/microsoft.public.access.forms/2006-07/msg00290.html>

- *From:* SteveS <sanfu@xxxxxxxxxx>
 - *Date:* Thu, 06 Jul 2006 23:37:13 -0800
-

"I wondered all night where the sun went when it set, and finally it dawned on me"

Hank,

So if I understand right, the variables that begin with "sys" are globally defined variables. You are using them in a function declared Public.

Trying to DIM variables in a public function when the variables are already DIM'd will generate an error (of course!!). OK, I got it. (better later than...)

For the SQL string, text needs to be delimited with quotes:

```
strWhere = "[NtSysCd] = "" & [sysNtSysCd] & "" And "
```

Dates are delimited with hashes (#):

```
strWhere = "[dteSysDate] = #" & Me.SomeDate & "# And " _ ...
```

Numbers do not require delimiters

```
strWhere = "[lngCd_ID] = " & Me.A_Number_ID & " And " _ ...
```

And....., its great you were able to find the problems and get it to work. I also struggled with creating the SQL string when I first started. It does get easier with practice. <grin>

Steve S.

"Veni, Vidi, Velcro"
(I came, I saw, I stuck around.)

Hank@ieultd wrote:

OK..... I dropped the Dim's and went back to the original sequene of events. I recompiled

Re: Multi Field SQL Where Clause

the project and discovered another error in two undefined entries. Once corrected and recompiled IT WORKS EXACTLY LIKE I WANT IT TOOOO!!!

Thanks Again for Your Help Determining the SQL Statement errors. By the way, is there any difference in the stating of integers/long & single fixed fields when addressing them in and SQL statement. I know the alpha/symbol variances but I am not sure if the different numeric types require any special tweaking within and SQL statement.

"Hank@ieultd" wrote:

Steve...Per the below:

The working variables starting with sys are publicly defined because I need them to be available to all the forms in this application stream. If a user elects to enter/view/delete a note at the PO header/PO Detail/Receiving Header or Detail I want them to use the same Public Function to display the Form with data from the notes table based on the key field info indicated. The Notes table fields which are also the controls on the NtMaint form are the NtSysCd/NtAplCd/NtCoNo/NtDvNo/NtLcCd/NtApLkNo/NtApLkLn/NtLn... the associated sys fields are dynamically refreshed by a private routine assoicated with a text button (not command button) on the form from which the user requests to view any associated notes. Once Refreshed the DisplayNotes Function is called/executed and should present and allow editing/adding/deletion of only the records from the notes table on the notes forms for the application form from which the user chose the notes display function. This is a simple treed architechure normalized data base with each application using the link table function to access only the necessary files for that application. The Notes function provides a pool of all notes for all applications through which a user can scan and then backward tree back to the actual application detail to which the note is attached.

I hope this helps. When the access application is executed I still get the same results I did with my original select/filter statement. I also created the DIM statements as you indicated and that didn't make any difference.

Thanks for Letting Me bang your brain around!!!

hank

"SteveS" wrote:

Also, add a line below the line

```
Public Function DisplayNotes()
```

```
' add this line
```

```
Dim sysform As String
```

```
Dim sysNtSysCd As String
```

```
Dim sysNtAplCd As String
```

```
'this was 'NotesSelectCriteria'
```

```
Dim strWhere As String
```

Re: Multi Field SQL Where Clause

'-----snip-----'

Another question....

Have you tried compiling the code?

Steve S.

"Veni, Vidi, Velcro"
(I came, I saw, I stuck around.)

SteveS wrote:

Hank,

Yes, it should have been a single (double) quote, not three (double) quotes.
And the MsgBox should have been "MsgBox strWhere".

The variables should be Dim'd. Public makes them available to every module.

```
sysNtCoNo  
=  
sysPUOHCNo  
sysNtDvNo  
=  
sysPUOHDvNo  
sysNtLcCd  
=  
sysPUOHLcCd  
sysNtApLkNo  
=  
sysPUOHNo
```

In this section of code, what/where are

```
sysPUOHCNo  
sysPUOHDvNo  
sysPUOHLcCd
```

Re: Multi Field SQL Where Clause

sysPUOHNo

Are they control names? On a form?

Where is this function? Code behind a form or a standard module? If a form, what is the form name?

How are you calling the function? (a button?)

Why are you using a Function instead of a Sub?

Have you traced (stepped thru) the code? Where is the error?

What exactly doesn't work?

Are the fields

[NtCoNo]

[NtDvNo]

[NtLcCd]

[NtApLkNo]

[NtApLkLn]

in the recordsource for the form "NtMaint"?

Sorry for all the questions, but since I can't see your mdb, I have to be a two year old.....
<grin>

Hang in there...

Steve S.

"Veni, Vidi, Velcro"
(I came, I saw, I stuck around.)

Hank@ieultd wrote:

Hello Again Steve;
I haven't given up yet of

Re: Multi Field SQL Where Clause

course so here goes.

The triple quotes cause a syntax error. When I remove the two additional quotes the syntax error goes away but the functionality is still absent. Do the definitions have to be DIM rather than PUBLIC as in Public sysNtCoNo as Single Public sysNtDvNo as Single Public sysNtLcCd as Single Public sysNtApLkNo as Long

etc. Note the co/dv & lc are single fixed fields. I don't know if that makes a difference with the quotes within the string or not. Also note, when using the string as you have defined it the correct values are displayed so it has to be close to correct. Just can't identify the obvious probably.

Thanks Again for all your help!!!
hank