

Re: Retrieving data from user input.

Re: Retrieving data from user input.

Source:

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.formscoding/2007-10/msg00249.html>

- *From:* Yepp <Yepp@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Wed, 3 Oct 2007 23:08:01 -0700
-

I am getting a compile error (method or data member not found) and also the "item not found in this collection" error. I ran the query by itself to test it and it worked fine. I checked the names to make sure they were correct...that is what is puzzling.

The names of the controls are Pro Number, Pro Last Name, and Pro First Name. The field names from the table are Pro_Number, Pro_Last_Name, and Pro_First_Name.

"Carl Rapson" wrote:

What errors are you getting? Are you certain you're using the correct names for the controls and the table fields? Did you try running the query standalone? What does your SQL string look like in the Immediate window?

What are the control names (the Name property in the Properties window for each control) and the field names (from the table)?

Carl Rapson

"Yepp" <Yepp@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message <news:24A407A5-2FA9-4004-B32F-59F7543185F2@xxxxxxxxxxxxxxxxxxxx>

Carl,

I put breaks in the code and keep getting errors on the lines in the if statement:

```
Me.txtField1 = rs.Fields![field1]
Me.txtField2 = rs.Fields![field2]
```

I changed the fields to:

```
Me.Pro_Last_Name = rs.Fields![Pro_Last_Name]
Me.Pro_First_Name = rs.Fields![Pro_First_Name]
```

Is my syntax off, or am I missing something? I'm using Me.<the text box's

Re: Retrieving data from user input.

control name> = rs.Fields![the field name of the text box].

"Carl Rapson" wrote:

Put a breakpoint in your code and step through each line, checking the values in the Immediate window. Is the AfterUpdate event actually firing (if not, you'll never reach your breakpoint)? What does your SQL string look like after you build it? Is the recordset actually being populated? Are there values in the recordset fields? One thing I do frequently is print the SQL string in the Immediate window, then copy the SQL statement and paste it into an empty Query window (in SQL view). If the query doesn't run, I can usually catch some errors there.

One important question: is the data type of the Pro_ID field (in the table) text or numeric? If it's text, even if you're storing a number you'll need to put quotes around the value in the SQL string you build:

```
strSQL = "...WHERE [Pro_ID]='" & Me.<your control name> & """
```

Note the single quotes around the value of Pro_ID. For clarity, here's the code with spaces inserted:

```
WHERE [Pro_ID] = ' ' & Me.<your control name> & ' ' "
```

Carl Rapson

"Yepp" <Yepp@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message
news:16E0173E-1884-40F3-9E1F-8AF0EE70F08D@xxxxxxxxxxxxxxxxxxxx

Carl,

I used my control name instead of the

Re: Retrieving data from user input.

'txtPro_ID' but still nothing happens.
There are no errors when I run the code...just none of the info fills in.

"Carl Rapson" wrote:

Is the control (not the field) really named 'txtPro_ID'? If not, use your own control name there.

Carl Rapson

"Yepp"

<Yepp@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

wrote in message

news:3F8052BE-C32A-418D-B482-0B2C81BC23E3@xxxxxxxxxxxxxxxxxxxx

Carl,

I tried your way (see below) but something is not working right. I keep getting a compile error on the strSQL line. Am I doing something wrong?

```
=====
Private Sub
Pro_ID_AfterUpdate()
Dim strSQL
As String
Dim rs As
DAO.Recordset
strSQL =
"SELECT
[prov1].LAST_NAME,
[prov1].FIRST_NAME
FROM
```

Re: Retrieving data from user input.

```
[prov1]
WHERE
[Pro_ID]="
&
Me.txtPro_ID
Set rs =
CurrentDb.OpenRecordset(strSQL)
If
rs.RecordCount
> 0 Then
Me.txtLAST_NAME
= rs.Fields![
LAST
NAME]
Me.txtFIRST_NAME
=
rs.Fields![FIRST
NAME]
End If
rs.Close
Set rs =
Nothing
```

=====
"Carl
Rapson"
wrote:

"Yepp"
<Yepp@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
wrote
in
message
news:BEA66879-28E7-4BCF-A8AD-12E83DD49E59@xx

I
have
a
form
that
has
several
fields,
one
of
which
is
called
Pro_ID.
I
would

Re: Retrieving data from user input.

like
to
use
whatever
number
is
entered
in
that
field
to
run
a
query
that
fills
in
the
rest
of
the
fields
with
information
associated
with
that
Pro_ID.
Any
ideas
as
to
how
I
can
achieve
this?
Thanks.

In
the
AfterUpdate
event
of
the
Pro_ID
control,
construct
an
SQL

Re: Retrieving data from user input.

statement
to
retrieve
the
values
you
want
and
place
them
in
the
appropriate
controls:

```
Dim  
strSQL  
As  
String  
Dim  
rs  
As  
DAO.Recordset  
strSQL  
=  
"SELECT  
field1,field2  
FROM  
table  
WHERE  
[Pro_ID]="  
&  
Me.txtPro_ID  
Set  
rs  
=  
CurrentDb.OpenRecordset(strSQL)  
If  
rs.RecordCount  
>  
0  
Then  
Me.txtField1  
=  
rs.Fields![field1]  
Me.txtField2  
=  
rs.Fields![field2]  
End  
If  
rs.Close
```

Re: Retrieving data from user input.

Set
rs
=
Nothing

Alternately,
you
could
use
a
series
of
DLookup
calls
to
do
the
same
thing,
although
using
multiple
DLookUps
usually
contributes
a
lot
more
overhead
(and
is
slower)
than
using
a
single
recordset:

```
Me.txtField1
=
DLookup("field1","table","[Pro_ID]="
&
Me.Pro_ID)
Me.txtField2
=
DLookup("field2","table","[Pro_ID]="
&
Me.Pro_ID)
```

Carl
Rapson

Re: Retrieving data from user input.