

## RE: Making one field on a form dependent on another fields results

*Source:* <http://www.tech-archive.net/Archive/Access/microsoft.public.access.forms/2005-03/2709.html>

---

*From:* Sprinks ([Sprinks\\_at\\_discussions.microsoft.com](mailto:Sprinks_at_discussions.microsoft.com))

*Date:* 03/22/05

Date: Tue, 22 Mar 2005 12:57:07 -0800

No, hang in there. It's always darkest before the dawn...

First of all, your approach to creating these mini-tables that consist of a numeric primary key and a text field is a good one. If County and State are a property of an Organization, rather than a Contact, though, they should be fields in the Organization table.

I'm not sure I understand what you mean by "go to the Fire Chief's database". What I think you mean is that you would like to be able to view and edit only the Fire Chiefs at some point in time, or to work with some other subset of the total records. This is done by either a Filter or a Query. Forms can be based either directly on a table, or on a query that might have more than one related table, and might have specific selection criteria. Or, you can dynamically set their filter programmatically by changing the form's Filter property.

One easy way to do this is to include a combo box in your form header to select the subset of records you'd like to work with. Do not bind it to a field. Then use the AfterUpdate event procedure to filter the records based on your selection:

```
Me.Filter = "Position = " & Me!MyComboBox  
Me.FilterOn = True
```

If you've never entered an event procedure, right click on the control and choose Properties, click on the Event tab, click in the AfterUpdate field, click on the ellipsis, and choose Code Builder. Access will create the shell of the procedure for you. Then enter the two lines above between the Sub...End Sub lines. Save and exit. The AfterUpdate field should now display [Event Procedure], letting you know there is one defined for this control and that it will execute when the AfterUpdate event occurs (whenever you change the value).

I hope this makes sense to you. Access can store up to 2 billion records, so it's not likely you'll be limited by keeping your Contacts in one table. It will make your life easier, not harder.

## Sprinks

"dawnykins" wrote:

> *I think I need to go back to the beginning, because now I am really  
> confused. I understand what your saying but I don't think I have the  
> knowledge to carry this out yet. I can say this.*  
>  
> *I have my first table, which is my contacts table listed with the fields I  
> want in it and the primary key code is a autonumber that is assigned to each  
> new record. I have another table that has autonumbered each position I have  
> entered in there. I have done this for tables named: position (the one were  
> talking about), county, organization, and State. So in my contacts form I  
> have the same fields as my table does and a few combo boxes that have drop  
> down menus for Position Title, Organization name, State and County names. I  
> wanted to set it up so that if the drop down menu is a position and we enter  
> that this person is a fire cheif, I wanted this information to go to the Fire  
> Chief's database.*  
> *I think your saying I don't need to do all this. Should I just keep the one  
> table and form then. Won't that get real big. I am trying to locate someone  
> in my very small town I live in that knows Access and can help a bit because  
> I am confused and this book I have isn't helping too much and it's hard to  
> understand what your saying. I afraid of putting you through too much trouble  
> to help me and not understand it.*

>  
> "Sprinks" wrote:

>  
>> *OK, before addressing the combo box, and how to display the information  
>> you're after, you're making this much too difficult. You do not need  
>> separate tables for Contacts, PoliceOfficers, Professors, Restaurateurs,  
>> etc.--you only need to distinguish between them with a different value in a  
>> field. You can then choose any one or several of them easily by selection  
>> criteria within a query.*

>>  
>> *The first step in database design is defining your "Things" (Tables) and  
>> their "Properties" or "Attributes" (Fields). Assign a numeric primary key to  
>> each table as a unique record identifier. I always, and other application  
>> developers usually, use an AutoNumber field type for the primary key.*

>>  
>> *So far, you've defined two things, Contacts and Positions. They are related  
>> tables, in that each Contact has a Position. The way to specify a position  
>> for each contact is to use a NUMERIC field for the PositionID that  
>> corresponds to Positions' primary key, called a Foreign Key. By linking the  
>> two tables in a query by the common field, the Foreign Key metaphorically  
>> "unlocks" the Positions table such that you have access to any of its fields  
>> for displaying on a form, or printing on a report. This allows you to  
>> \*store\* an efficient numerical value, yet display the more meaningful data.  
>> And there's one more big advantage to this approach.*

>>  
>> *Say you have an Orders table, and you store the name and address fields in  
>> your Orders table. If your customer changes their name or address, you've*

microsoft.public.access.forms: RE: Making one field on a form dependent on another fields results

> > got to change every Orders record for that customer. Moreover, if you make a  
> > typo, you might end up with something like the following:  
> >  
> > Very Large Company  
> > VeryLarge Company  
> > Very Lrg. Co.  
> >  
> > all of which are the same customer, but stored under different names. If,  
> > however, you store only the customer number in the Orders table and a  
> > customer moves or changes their name, you change a single record in the  
> > Customer table and you're done.  
> >  
> > Getting back to your example, your Positions table should be structured like  
> > this:  
> >  
> > PositionID AutoNumber (Primary Key)  
> > PositionName Text  
> > ...any other Position-specific data like salary-class, etc.  
> >  
> > and your Contacts table should have a numeric field for the Position.  
> > However, noone wants to have to remember a code, therefore, you create a  
> > combo box that displays a list of names, but \*stores\* the related code in the  
> > underlying field. They work via these key properties:  
> >  
> > RowSource SQL for the fields to include in the combo box. These may be  
> > visible or not. Each field is a column of the combo box.  
> >  
> > BoundColumn The index (1, 2, ...) of the column to store in the  
> > ControlSource.  
> >  
> > ControlSource The field of the form's underlying table in which to store  
> > the BoundColumn.  
> >  
> > ColumnWidths How much horizontal space to allow for display of the column.  
> > If a column width is set to 0", it does not display at all. After the user  
> > makes a selection, the first non-zero-width column is what remains showing in  
> > the box.  
> >  
> > In most instances, you don't really care about fields like the PositionID  
> > code; it's just an efficient storage means. What's normally important is  
> > displaying meaningful text. To do this in your situation, alter the Row  
> > Source and Column Widths properties:  
> >  
> >  
> > Row Source: SELECT Positions.ID, Positions.[Position Title] FROM Positions  
> > ORDER BY Positions.[Position Title];  
> > Column Widths is: 0";x" (where x is a width arrived at by trial and error)  
> > Bound Column is: 1  
> > Control Source is: Position (make this a numeric field)  
> >  
> > When a selection is made, the PositionTitle will display, and the code will

microsoft.public.access.forms: RE: Making one field on a form dependent on another fields results

> > *be stored in the numeric field Position.*  
> >  
> > *Now, if you want to show the code for informational purposes, simply add a*  
> > *textbox with its control source set to the Position field.*  
> >  
> > *Another way to do it, and this would apply to other combo box cases where*  
> > *you wish to display other columns of the selected row, is to use the Column*  
> > *property. Unfortunately, the index for this property begins with 0. To*  
> > *display the third column of a cbox, you'd set the textbox' Control Source to:*  
> >  
> > `=Me!MyComboBox.Column(2)`  
> >  
> > *Hope that helps.*  
> > *Sprinks*  
> >  
> >  
> > *"dawnykins" wrote:*  
> >  
> > > *Box 1*  
> > > *Row Source: SELECT Positions.[Position Title], Positions.ID FROM Positions*  
> > > *ORDER BY Positions.[Position Title];*  
> > > *Column Widths is: Blank*  
> > > *Bound Column is: 1*  
> > > *Control Source is: Position (this is the name of my field)*  
> > >  
> > > *Here's what I am trying to accomplish with my database. I want our office to*  
> > > *be able to enter a person's name in the database (by the form I call*  
> > > *Contacts). From that point I want the information to go to one table with*  
> > > *everything, (names addresses, position name....etc..) and if a person is a*  
> > > *police officer per say, I want that information also to go to a table for the*  
> > > *police officers. Same as another other position going to their perspective*  
> > > *tables. I think I have to have the position ID as the primary in all the*  
> > > *other tables in my contact table and form in order for the information to go*  
> > > *to each table. I am really new to this and all I have to go by is what I find*  
> > > *in this discussion and the book access for dummies. So any help is greatly*  
> > > *appreciated.*  
> > >  
> > > *"Sprinks" wrote:*  
> > >  
> > > > *Please post the RowSource, ColumnWidths, Bound Column, and ControlSource*  
> > > > *properties. Also, out of curiosity, why do you wish to display the code when*  
> > > > *you're already displaying more meaningful text?*  
> > > >  
> > > > *Sprinks*  
> > > >  
> > > > *"dawnykins" wrote:*  
> > > >  
> > > > > *Thank you Sprinks I appreciate it. I have searched and I guess I am not*  
> > > > > *getting the right question in there or something. What I have is this: One*  
> > > > > *combo box which gives me a choice of choosing the positions I want. By*  
> > > > > *positions I mean what a person does for a living. Then I want another box to*

microsoft.public.access.forms: RE: Making one field on a form dependent on another fields results

>>>> *bring up the ID number that is associated with that position that I have  
>>>> chosen. Does this make sense. This position ID number is the primary key in  
>>>> many of my tables. I do not want this second box to be a combo box. Is this  
>>>> possible? I can't find any specifics in my searching. Also, I can't seem to  
>>>> get this code thing down in VBA. Your help is greatly appreciated.*  
>>>>  
>>>> *"Sprinks" wrote:*  
>>>>  
>>>> *> It depends on what you're trying to do. If you want to display a calculation  
>>>> > in a form control, such as an extended price dependent on the quantity and  
>>>> > unit prices, set the control's ControlSource to a valid expression, such as:  
>>>> >  
>>>> > =[Qty] \* [UnitPrice]  
>>>> >  
>>>> > 99.999% of the time, there is no need to store this value in your table,  
>>>> > since it can be calculated on-the-fly in a query. To do so also requires VBA  
>>>> > code, since a ControlSource can either be a fieldname, in which case data  
>>>> > entered in the control is stored in the field OR a calculation, but not both  
>>>> > simultaneously.*  
>>>>  
>>>> *> If you wish to limit the choices of a downstream combo box by a prior field,  
>>>> > you change the former's Row Source as appropriate. For example, once having  
>>>> > chosen Region, you might want to limit the SalesRep field to those in the  
>>>> > region:*  
>>>>  
>>>> *> Me!MyComboBox.RowSource = "SELECT SalesRepID, SalesRepName FROM SalesReps  
>>>> > WHERE [Region] = " & Me!MyRegionControl  
>>>> >  
>>>> > By the way, your question is a frequent one. Be sure that you understand  
>>>> > the distinction between a field, which exists in a table, and controls that  
>>>> > exist on forms, which may or may not be bound to fields in the form's  
>>>> > underlying RecordSource.  
>>>> >  
>>>> > Hope that helps.  
>>>> > Sprinks  
>>>> >  
>>>> > *"dawnykins" wrote:*  
>>>> >  
>>>> > *> How do you make one field on a form dependent on what another field's results  
>>>> > > are?**