

RE: linking data from one table to another

Source:

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.gettingstarted/2008-06/msg00052.html>

- *From:* Ken Sheridan <KenSheridan@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Tue, 3 Jun 2008 04:45:00 -0700
-

Firstly I'd suggest you make sure the four tables are set up correctly before tackling the form. Getting the 'logical model' right is the key to a successful database. Designing the interface then becomes pretty straightforward.

The Clients, Employees and Projects tables (or whatever you choose to call them) are easily set up as they are just lists of each, Clients having one row per client, Employees one row per employee and Projects one row per project. Give each an autonumber primary key column, ClientID, EmployeeID and ProjectID. Other columns in each represent their attribute types, such as the names of clients and employees, and project descriptions.

The WorkAllocations table, which models the relationship type between the other three is what pulls everything together. In this you should have three foreign key columns, ClientID, EmployeeID and ProjectID again, but this time not autonumbers, just straightforward long integer number data types. Other columns in this table will represent other attributes as I said, such as the date when a project is allocated to an employee.

Enter some data into the Clients, Employees and Projects tables. Lets say you have rows with the a ClientID of 1, an EmployeeID of 2 and a ProjectID of 3. Now enter a single row into WorkAllocations with a ClientID of 1, an EmployeeID of 2 and a ProjectID of 3. This row represents the fact that project 3 for client 1 has been allocated to employee 2.

Now create a query which joins these four tables, joining WorkAllocations to Clients, Employees and Projects on the ClientID, EmployeeID and ProjectID columns respectively. Add suitable fields such as the client's names and address, the employee's names and the project name or description to the design grid in query design view and then open the query. You should see a row with the relevant names etc for project 3, client 1 and employee 2. This shows your logical model is set up correctly.

As far as the form is concerned I notice that in my earlier reply I put tblEmployees rather than Employees as the table name in RowSource property for the combo box. This is because I took the example from a n earlier reply for someone who used the 'tbl' prefix for table names. I personally don't use such 'tags' so if you've called your table Employees then change the

RE: linking data from one table to another

RowSource so it uses that name. The column names FirstName and LastName used in the RowSource property should also match your real column names exactly. FullName is an alias given to the concatenated first and last names, not a column in the table, so that doesn't have to match anything.

I think you'll find using a combo box a lot simpler than a command button, so if you've got the logical model right, and the table and column names in the RowSource property match your actual ones there's no reason why it shouldn't work.

To show the employee contact details use computed controls, calling the DLookup function in the ControlSource as I described. Again the table and column names must match the real ones.

Clients and Projects are treated similarly, using combo boxes bound to the ClientID and ProjectID columns, and computed controls to show other values from those tables.

The form will be based on the WorkAllocations table of course, so you can't enter data directly into the Clients, Employees or Projects tables in the form. So if you want to enter a new client say then you have to switch to a separate clients form. There is a method for automating this, however, by using the combo box's NotInList event procedure, which enables you to type a new client name into the combo box and have this automatically open the clients form, returning you to the WorkAllocations form when you've entered the new client. The same goes for entering a new employee or project. I can help you with this once you've got the form working, but I'll need to see the RowSource properties of the three combo boxes so I can give you the right code. If you've not done any VBA coding before don't worry; its not as scary as you might think and I'll walk you through it.

If you are still having problems getting the form set up feel free to mail me at:

kenwsheridan<at>yahoo<dot>co<dot>uk

and I can send you a simple little .mdb file which wraps it all up.

Good luck,

Ken Sheridan
Stafford, England

"kimmieboot" wrote:

hi, thanks for the instructions, I have been trying for several hours to set this up as per your instructins, with no result. I have entered the data into the row source using your instructions as a guide, it has the drop down box when you go to the form, but there is no data in the drop down box.

RE: linking data from one table to another

What I would really like to happen is – have a command button that allows you to view a list of all the first names of employees, select the employee, then all the employee contact details are listed back on the form into the various fields ie first name, surname, address, phone number.

Are you able to provide any further assistance/help ?

—

thanks, kimmieboot

"Ken Sheridan" wrote:

What you have here is a 3–way relationship type between Clients, Employees and Projects, the last being the work in question. So you need three tables, one for each of these, each of which has columns representing the attribute types of each, e.g. FirstName, LastName etc for Employees. To model the relationship type you need a fourth table WorkAllocations say with foreign key columns which reference the primary keys of the other three tables, e.g. ClientID, EmployeeID and ProjectID. These can be autonumbers in the referenced tables, but should be a straightforward long integer number data type in WorkAllocations. You can also have other non–key columns in WorkAllocations to represent other attributes types, e.g. DateAllocated. Don't include other columns from the referenced tables in this table, however, e.g. don't include a client's address as this is available from the Clients table via the relationship on the ClientID columns.

In a form based on the WorkAllocations table you can use combo boxes bound to the ClientID, EmployeeID and ProjectID columns. The value of these will be the numeric IDs but you'll see the corresponding text values from the referenced table by setting them up as follows, e.g. for the EmployeeID combo box:

```
RowSource: SELECT EmployeeID, FirstName & " " & LastName AS  
FullName  
FROM tblEmployees ORDER BY LastName, FirstName;
```

```
BoundColum: 1  
ColumnCount: 2  
ColumnWidths 0cm;8cm
```

If your units of measurement are imperial rather than metric Access will automatically convert the last one. The important thing is that the first dimension is zero to hide the first column and that the second is at least as wide as the combo box.

To show other values from the referenced table you can use unbound computed controls, e.g. to show the selected client's address you'd set the

RE: linking data from one table to another

ControlSource property of an unbound text box to:

```
=DLookup("Address","Clients","ClientID = " & [ClientID])
```

With the above model you can allocate as many projects per client to as many employees as necessary. A single project for a client could be allocated to one or more employees for instance; or different projects for a single client can be allocated to the same or different employees. You could even have the same project undertaken for more than one client, though I'd guess that's probably not a realistic scenario.

Ken Sheridan
Stafford, England

"kimmieboot" wrote:

Hi, I have created a database in Access 2000 which has two tables. I created a form from the data of one table which has all contact details of new clients. It also has a section to add who the work has been allocated to. I have created another table with all employees (this is who the work is allocated to) I want to dump the information of the employees into the form – the only way I have been able to do this is drop down list for each field but this takes forever to go through them one by one with first name, last name, address and contact numbers, etc.

Is there anyway that I can dump this information in one go by clicking on one field in the other table ?????

Any help is most welcome as I have spent several days trying to do this !!!!

--
thanks, kimmieboot