

Calculated Fields in Queries and Forms

What is a Calculated Field?

Sooner or later you will want to perform calculations with your data, but calculations are not part of your data, they are something derived from it. They are not created as part of a table, but instead are performed in a query, form or report.

It is normally not good practice to store data that could be calculated from other stored data. This is because storing calculated data wastes storage space and stored calculations are "frozen" and will not respond to changes in the values from which they were derived.

Creating a Calculated Field in a Query

This simple example shows how you might store the details of a transaction in a table by entering the *Unit Price* of an item and the *Quantity* purchased, but not the resulting *Total* because this can be easily calculated from the existing data (Fig. 1).

ItemID	Description	UnitPrice	Quantity
1	Copy Paper A4 (ream)	£2.45	10
2	Laminating Sheets A5 150 gsm (100)	£12.75	2
3	Envelopes DL White (100)	£3.40	3

Fig. 1 Only the minimum required data is stored in the table.

To perform a calculation in a query you must create a new *Calculated Field*. Do this by entering an expression in the *Field* row of an empty column. The expression should consist of a **Name** for the new field followed by a **colon** then the **calculation** itself. The name should adhere to the usual naming rules for field names. The calculation will normally contain references to other fields along with the appropriate mathematical operators (+, -, *, / etc.). Do not use the percent sign (%) in your calculations because this symbol has a special meaning in SQL (it is equivalent to the asterisk wildcard).

To continue the simple example above, the *Total* value of the purchase is calculated in a query by creating a new, calculated field in which the *Unit Price* is multiplied by the *Quantity* (Fig. 2).

Field:	Description	UnitPrice	Quantity	Total: [UnitPrice]*[Quantity]
Table:	tblOrderItems	tblOrderItems	tblOrderItems	
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:				

Fig. 2 The calculated field multiplies two other fields.

Running the query displays the result of the calculation (Fig. 3).

Description	UnitPrice	Quantity	Total
Copy Paper A4 (ream)	£2.45	10	£24.50
Laminating Sheets A5 150 gsm (100)	£12.75	2	£25.50
Envelopes DL White (100)	£3.40	3	£10.20

Fig. 3 The query displays the result of the calculation.

Reusing Calculated Fields

Once created a calculated field behaves like any other field in that you can apply criteria to filter the data, sort by the calculated field, or use it in further calculations. This example uses the *Iif()* function (called *Immediate If* in Access and equivalent to Excel's *If* function) to apply a 5% discount to order items totalling more than £20 in value (*Fig. 4*).

Quantity	SubTotal: [UnitPrice]*[Quantity]	Total: Iif([SubTotal] > 20, [SubTotal]*0.95, [SubTotal])
tblOrderItems		
✓	✓	✓

Fig. 4 A calculated field uses values created by another calculated field.

Running the query displays the result of the calculations (*Fig. 5*):

	Description	UnitPrice	Quantity	SubTotal	Total
▶	Copy Paper A4 (ream)	£2.45	10	£24.50	23.275
	Laminating Sheets A5 150 gsm (100)	£12.75	2	£25.50	24.225
	Envelopes DL White (100)	£3.40	3	£10.20	10.2
*					

Fig. 5 The query displays the result of the calculations.

Formatting Calculated Fields

As with Excel, Access queries format the result of a calculation based on the format of the first field used in the expression. But sometimes, as in the previous example (*Fig. 5*), it can't ascertain which format to use so you have to set the format yourself. In query design view right-click on the new column and choose **Properties** to open the **Field Properties** dialog (make sure to click in the right place as clicking elsewhere opens the *Query Properties* dialog). Choose a pre-defined format from the list (*Fig. 6*) or create a custom format by entering the usual formatting codes.

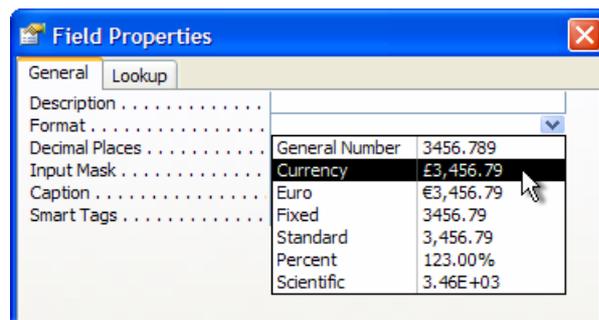


Fig. 6 Setting the format property of a field.

When run the query will display the calculated value in the specified format (*Fig. 7*).

	Description	UnitPrice	Quantity	SubTotal	Total
	Copy Paper A4 (ream)	£2.45	10	£24.50	£23.28
	Laminating Sheets A5 150 gsm (100)	£12.75	2	£25.50	£24.23
▶	Envelopes DL White (100)	£3.40	3	£10.20	£10.20
*					

Fig. 7 The calculated field now displays its result in the desired format.

Showing Calculations on Forms and Reports

The same principle is used to display calculated values on a form or report. Add an unbound textbox to your form or report (i.e. one that is not linked to a field) and type your expression directly into it. Unlike the query, the form or report does not require that you add a fieldname to your expression so you can omit the name and colon. Instead, prefix the expression with an equals sign (=).

If you find it difficult to write your expression directly into a small textbox open its properties window (right-click on the textbox and choose **Properties**) and type the expression into the textbox's *Control Source* property in the *Data* section of the properties window (Fig. 8). If you need even more space right-click on the Control Source textbox and choose **Zoom** to open a large box to type into.

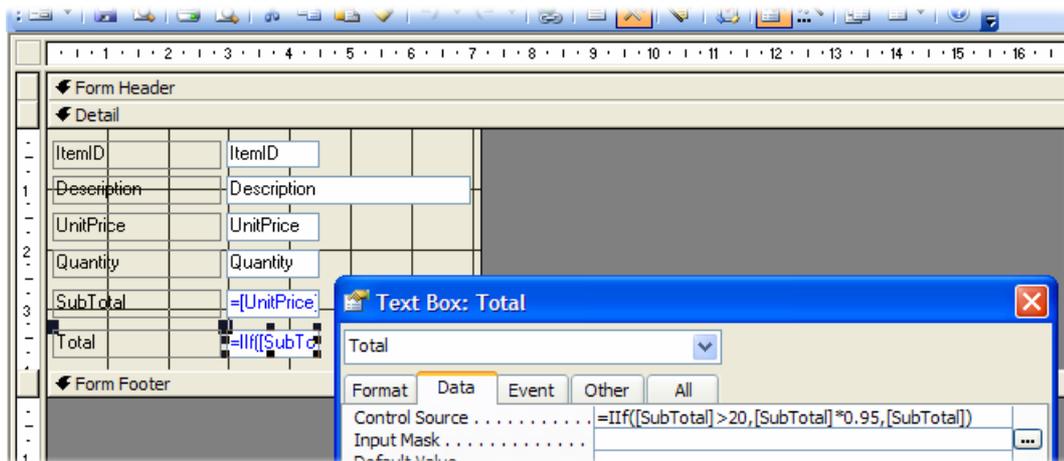


Fig. 8 Creating a calculated field on a form.

Access will have given your textbox a name such as "Text10" when you added it to the form or report. It is a good idea to change this to a more meaningful name (in this example I have named one *SubTotal*) so that your calculation expression makes more sense. Here is how the calculations look on the form (Fig. 9).



Fig. 9 A form displaying calculated values.

Naturally, the results displayed in calculated fields can not be edited by the user so there is little point encouraging them to enter the field. Use the textbox properties window to set the *Locked* property to "Yes" and the *Tab Stop* property to "No".

What Sort of Calculation Can Be Performed?

Calculations are simple and straightforward. You just enter field names instead of numbers. Very simple examples have been shown here, but calculations can be as complex as you need them to be. Access has a wide range of functions that you can include in your calculations. Use the **Build** tool to view a list of available functions and for help on how to use them (right-click in the **Field** row of the query design grid and choose **Build**).

This example demonstrates that the fields used in the calculation do not have to be included in the query design grid as long as they are members of the recordset being queried (Fig. 10):

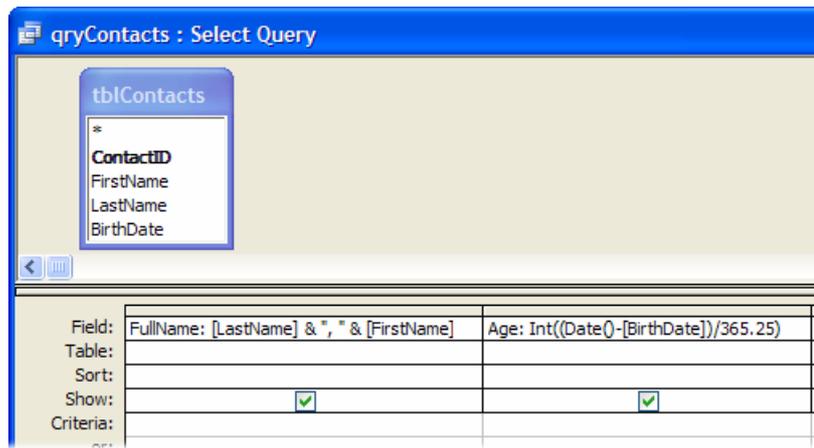


Fig. 10 A query consisting entirely of calculations.

A *FullName* field has been created by concatenating (joining together using the ampersand character (&)) the *LastName* and *FirstName* fields with a comma and a space in between. The *Age* field calculates the person's age from their *BirthDate* by subtracting it from the current date as supplied by the *Date()* function, then dividing the result by 365.25 (the average number of days in a year) and rounding down the result with the *Int()* function. Here's the result (Fig. 11):

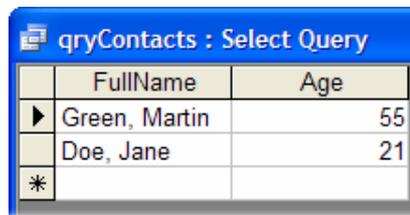


Fig. 11 The result of the query designed in Fig. 10

What Happens to the Calculated Data?

The data displayed in a calculated field is temporary and exists only when the query is run or the form or report is displayed. For this reason it occupies no storage space. The result is always up-to-date, reflecting the current source data.

If you need to take a snapshot of the calculation to preserve the calculated values there are many different ways to choose from. If you use a calculation in a **Make-Table** query the new table will include a new field containing the new calculated data. Alternatively use one of the many methods of exporting the data to a different format, such as an Excel workbook, a Word document or a text file.