

FAQ: Access Topics

This week's FAQ was: What's the difference between the SQL which is generated automatically when you create an Access query on the QBE grid and the hand-coded SQL "Select... " statement used in the example for the homework?

When you design any query on the Access QBE grid, you can click the little button on the upper left and obtain the SQL "translation" of the query you just created by using the grid. The reason for this is the "Jet Engine" which governs databases in Access or Visual Basic speaks a dialect of SQL. Remember that any SQL statement is simply text. As such, it may be cut and pasted and used elsewhere, which is sometimes a handy thing.

Normally, however, you design a query on the QBE grid for a specific purpose. For instance, you want all the records in a Customer table for "Buncombe County", and no other county. You do this so often, and find this particular query so useful, that you save it as, say, "Buncombe_Query". When you look at the "Queries" tab of your database, sure enough you will find a listing for "Buncombe_Query" in your query window.

When you designed "Buncombe_Query" on the QBE grid, you dragged all the fields for the table "Customers" onto the grid, found the column [CountyName], and under conditions, added: ="Buncombe". Later on, when you're writing code for this database, you can access your Buncombe recordset by doing the following:

```
dim MyBunk as recordset  
set MyBunk=Currdb.OpenRecordset("Buncombe_Query")
```

The Jet Engine looks for a table or query which has the name "Buncombe_Query". Since it's listed in your queries window, it finds it and opens the recordset.

But what if you didn't save this particular query in your database, and you still want the recordset in your code for some purpose? The OpenRecordset(...<whatever>...) method will look for ANY definition of a recordset between the "(" and the ")". If it isn't defined in your database by a named and saved table or query, it will assume you're speaking SQL. If you look at the SQL equivalent of your Buncombe query, it will look something like this:

```
"Select [ID], [CustomerName], [CustomerAddress],[County_Name] from  
Customers where [County_Name] = 'Buncombe' "
```

or, if you want all fields selected, use the "all fields" notation: " * "
thus:

```
"Select * from Customers where [County_Name] = 'Buncombe' "
```

The latter is the SQL equivalent text of your Buncombe_Query.

So, if you don't have a Buncombe_Query in your database (say it's a new Access base and you didn't feel like importing the query) you can accomplish the same result if you say:

```
dim MyBunk as recordset
set MyBunk = Currdb.OpenRecordset("Select * from Customers where
[County_Name] = 'Buncombe' ")
```

I call this technique the "SQL on-the-fly" query.

Why not just define the query, add it to your database, and use OpenRecordset("Buncombe_Query") ? The answer is that the on-the-fly query may be modified at execute time. In most data processing, your task is not to write a report which is good only for Buncombe County, but chances are you will get a raw database update which contains a new customer table for an unpredictable number of counties. You must produce results for ALL counties, one after the other. That means you must "hard-code" a bunch of QBE queries in advance, one for each county. This is dismal. On the other hand, if you do this:

```
...
dim ThisCounty as String
...
ThisCounty = "Saratoga"
...
set MyCounty = Currdb.OpenRecordset("Select * from Customers where
[County_Name] = '" & ThisCounty & "'")
...
```

ThisCounty can contain the name of ANY county. (In this case, the county is "Saratoga", but it's essentially the same query.) Usually, you obtain a unique list of all the counties in your raw base. Then take this list, and equate each county name in turn with ThisCounty and run your SQL on-the-fly query as many times as there are counties in the list. You would do this in a loop, where each trip through the loop would load a different county name into ThisCounty.

For this assignment, we're only illustrating an on-the-fly query which selects a "hard-coded" criterion for a particular county, but later we'll be substituting a **variable** for the name of the county.

Friday, I'll go over SQL "Select..." queries grammar. Hint: they all look alike, viz.:

```
" SELECT {... <all fields>...or ...<a list of specific fields>...} FROM
{ <a table or query>} WHERE { <...a particular field..> = < ...definite
value...> } "
```