

# Null Responses in Parameter Queries

## What If the User Ignores the Prompt?

### Parameter Queries

The idea of a parameter query is that it offers the user some choice when they run the query. Instead of you having to anticipate the various combinations of criteria that you are going to need and creating a separate query for each, you use parameters to prompt for information when the query is run. Access does this by presenting the user with an input box into which they type what they want to see. When they click the **[OK]** button the query places what the user typed into the appropriate place in the query definition and runs the query.

But what if the user leaves the input box empty? You might expect that if the query receives no input it would return all the records, but that isn't what happens. It returns nothing at all - an empty recordset.

It's really easy to adjust your criteria so that the query will return *all* the records when the user ignores the prompt and doesn't type anything. They just click **[OK]** and if they want to see all the records. Here's how...

Field:	Office	Department
Table:	Staff	Staff
Sort:		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	[Which Office?]	
or:		

Fig. 1 A regular parameter query.

### A Regular Parameter Query

Here's a regular parameter query with a prompt for the user to enter the name of the Office whose records they want to see in the query's result (Fig. 1).

In this example the user typed **Cardiff** and got the following result you see on the left (Fig. 2).

If they had ignored the prompt and left the input box empty they would have seen no results at all.

Office	Department
Cardiff	Management
Cardiff	Production
Cardiff	Shipping
Cardiff	Sales
Cardiff	Production
Cardiff	Production

Fig. 2 The result of the parameter query when the user entered "Cardiff".

### Giving the Option to Return All Records

Suppose the user doesn't know what to type, or perhaps this time would like to see all the records? All we have to do is adjust the criteria to accept whatever the user types, or to return all the records if they type nothing (i.e. if the input is "null").

Here the criteria have been modified to accept a null entry (Fig. 3).

Field:	Office	Department
Table:	Staff	Staff
Sort:		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	[Which Office?] Or Like [Which Office?] Is Null	
or:		

The parameter contains an "Or" clause enabling the query to interpret a null response as a request to see all the records.

Fig. 3 A parameter query modified to accept a null entry.

Office	Department
London	Management
Birmingham	IT
Berkeley	IT
London	Production
London	Production
London	Research

Fig. 4 The result obtained when the user left the prompt box empty.

...and here's what the user saw when they left the prompt box empty (Fig. 4).

Ignoring the prompt returns all the records thanks to the modification to the criteria. To summarise...

A parameter that requires an input from the user, otherwise no records are returned is written...

#### [type prompt here]

A parameter that can accept an input from the user, but that will return all records if no input is made is written...

#### [type prompt here] Or Like [repeat prompt here] Is Null

As you can see, the prompt is entered twice although the user sees only one input box. It is important that the prompt is exactly the same in both cases, otherwise Access will treat them as separate parameters and the user will see two input boxes (although the query will *probably* still work!).

#### Variations on a Theme

If you prefer you can put the two parts of the parameter on separate lines in the query grid (Fig. 5).

Field:	Office	Department
Table:	Staff	Staff
Sort:		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	[Which Office?]	
or:	Like [Which Office?] Is Null	

Fig. 5 The "Or" clause can be written on a separate line.

Going down the column is equivalent to typing "Or" in the criteria. But I prefer to do it the other way. Doing it this way is fine if you aren't combining the parameter with criteria on other fields. If you are, then you have to be careful that your criteria read the way you intended. I find it's easier to put the whole thing on one line.

*TIP: Sometimes you have a lot to type in the cell of the QBE grid. You can stretch the column to fit what you type - point to the tops of the columns where they meet. When your cursor changes from an arrow to a black cross with horizontal arrows you can drag to the desired width or double-click to snap to fit. An easier way is to right-click on the cell and choose Zoom... from the shortcut menu. A large text box opens into which you can type your entry. Close the text box to put your typing into the cell. This is great for doing those fiddly corrections to existing entries. It works in tables too!*

Just like regular parameters, these ones can be combined into multiple parameters. For example, I could have had an additional prompt for *Department* which could also accept a null entry (Fig. 6).

Now the user can specify an *Office* or a *Department* or *neither* or *both*. Now don't tell me you don't think that's clever!

Field:	Office	Department
Table:	Staff	Staff
Sort:		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	[Which Office?] Or Like [Which Office?] Is Null	[Which Department?] Or Like [Which Department?] Is Null
or:		

Fig. 6 A query with two parameters, each of which can accept a null entry.