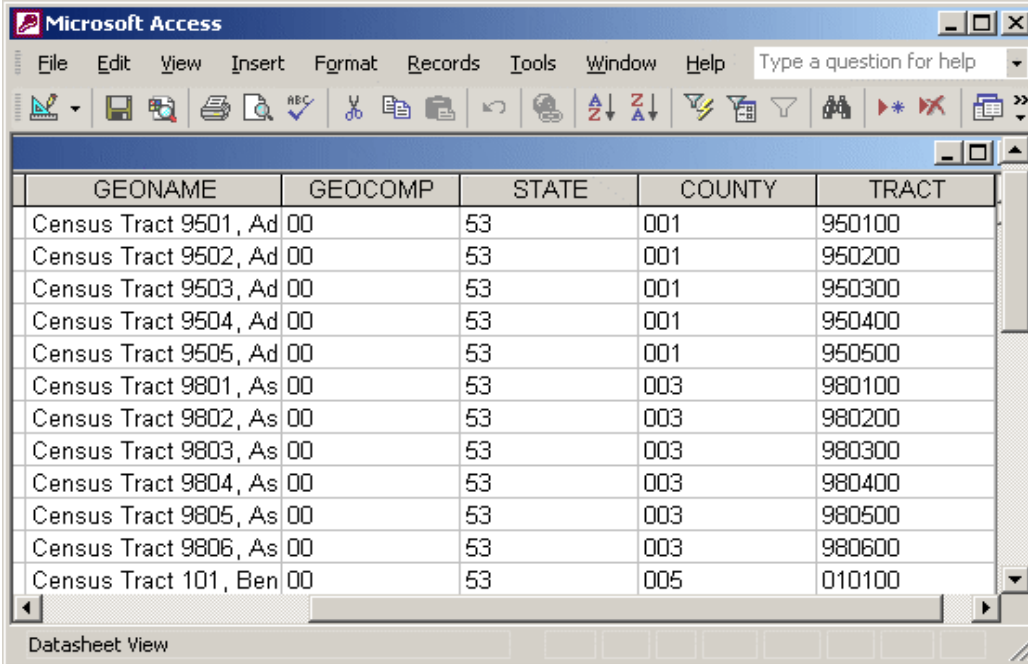


How to Concatenate Cells in Microsoft Access

This tutorial demonstrates how to concatenate cells in Microsoft Access. Sometimes data distributed over multiple columns is more efficient to use when combined into one column. For example, when downloading Census data from the American Factfinder website, number codes used to identify geographic entities such as state, county, and census tract may be presented in separate columns. In order to join this table to another table or boundary file, you may need to combine the columns in the data table so that they match the ID column in the other table or file (which would then allow you to join the two). The example below in Figure 1 is a table called *GeogTract* stored within an Access geodatabase:

Figure 1



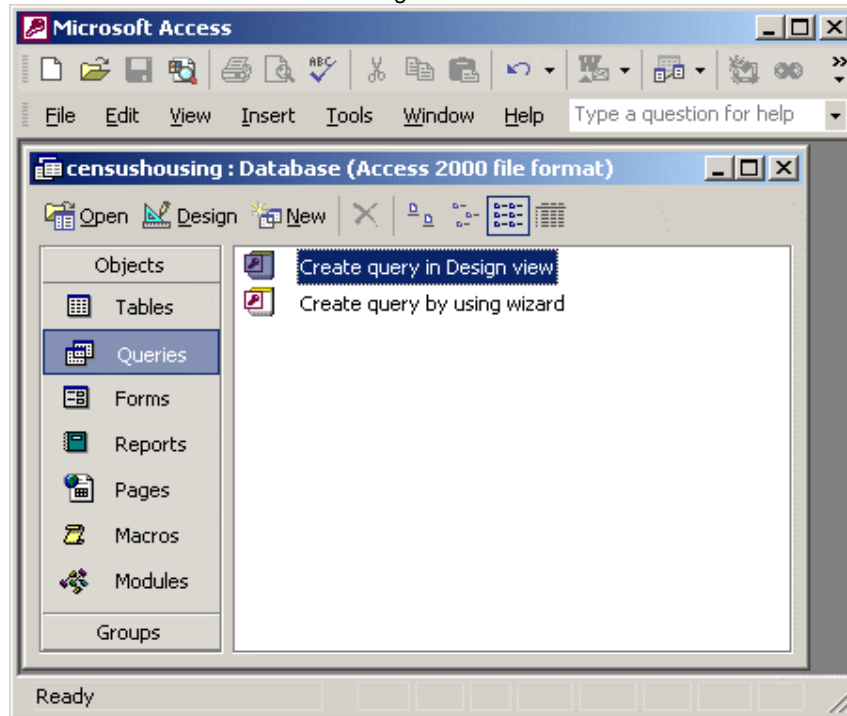
GEONAME	GEOCOMP	STATE	COUNTY	TRACT
Census Tract 9501, Ad	00	53	001	950100
Census Tract 9502, Ad	00	53	001	950200
Census Tract 9503, Ad	00	53	001	950300
Census Tract 9504, Ad	00	53	001	950400
Census Tract 9505, Ad	00	53	001	950500
Census Tract 9801, As	00	53	003	980100
Census Tract 9802, As	00	53	003	980200
Census Tract 9803, As	00	53	003	980300
Census Tract 9804, As	00	53	003	980400
Census Tract 9805, As	00	53	003	980500
Census Tract 9806, As	00	53	003	980600
Census Tract 101, Ben	00	53	005	010100

Notice how the state, county, and tract codes are stored in different columns. In this tutorial, we will combine these three columns into one. Unlike Excel, which allows you to insert a new, calculated column directly into a spreadsheet, in Access we must use the query builder to create a whole new table that will include our calculated field. This may sound daunting, but it is actually quite easy to do. Queries are specific views that we create for tables. We can create several different kinds of queries that can: display certain records in a table based on some criteria, create summary columns, aggregate data, average data, and bring data from two or more tables together into one view. In this example, we will create a query that will combine, or concatenate, several columns into one.

Building The Concatenation Query

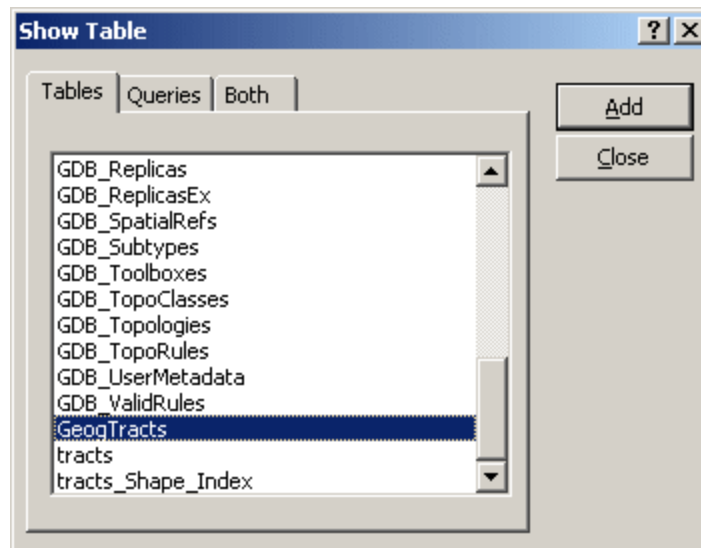
The first step is to close the table, and move from the *Table Objects* view (which lists all of the tables in the database) to the *Query Objects* view, by clicking *Queries* in the menu on the left (see Figure 2). Since we have not built any queries, there will not be any listed in the objects window. Click on *Create Query in the Design* view to enter the query builder window.

Figure 2



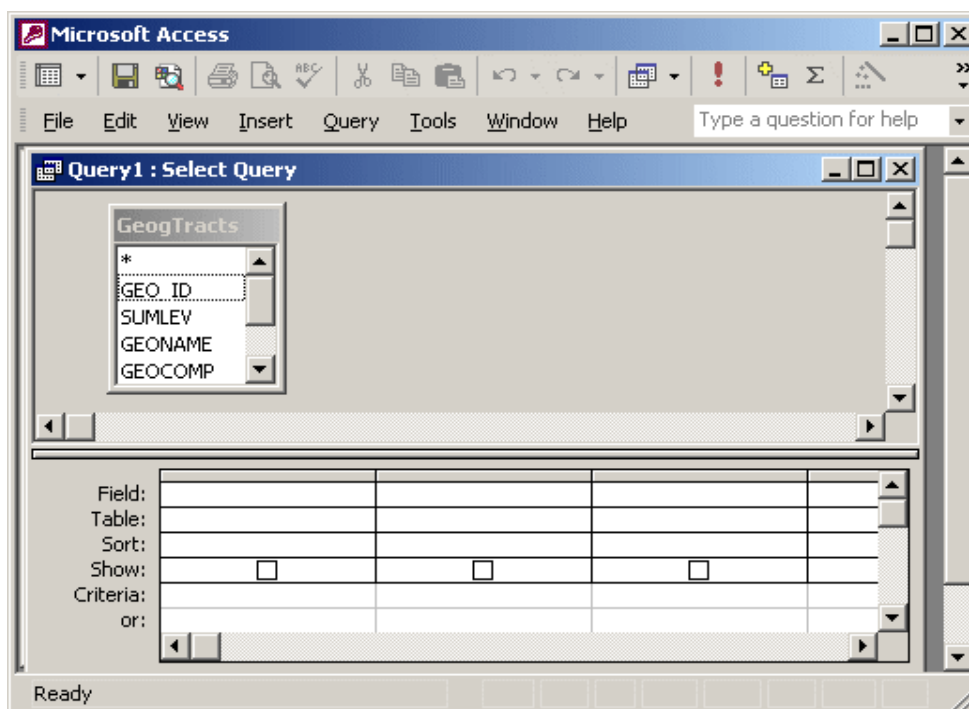
The first question we will be asked is: what tables do you want to add to the query (Figure 3)? Scroll through the list of tables and select the relevant one – in this case, we select the *GeogTracts* table - click *Add*, and then click *Close*.

Figure 3



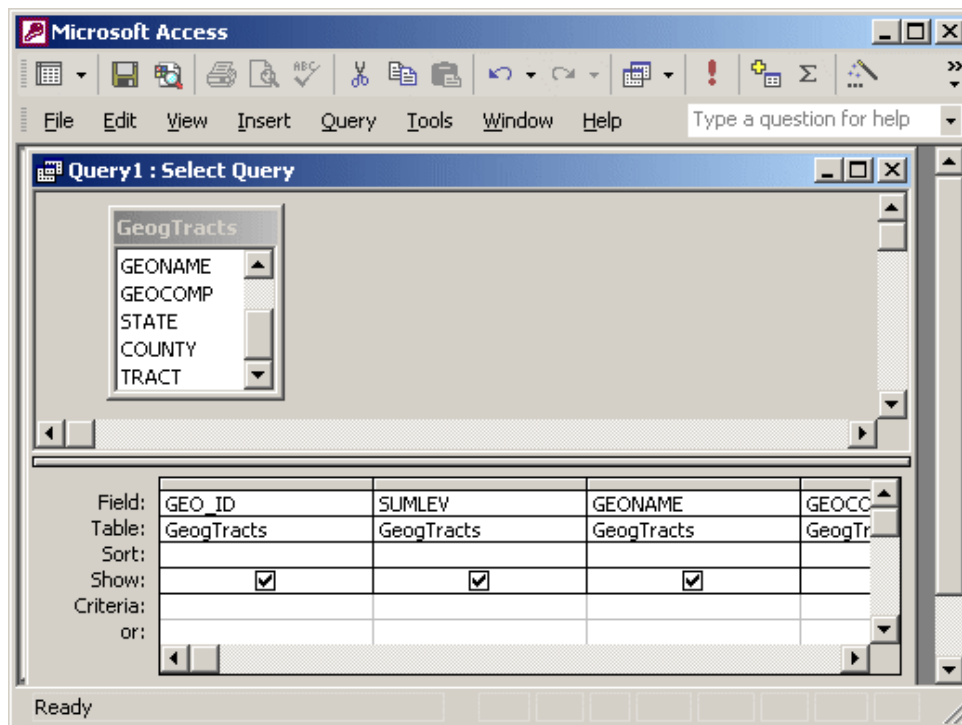
Once we make our selection, we are presented with the query design view window. At the top left is the *GeogTracts* table that we added, with a list of all the fields (columns) in the table. At the bottom of the window is a series of blank boxes that represents the fields from the table that we want to add to the query (see Figure 4). At this point, they are blank.

Figure 4



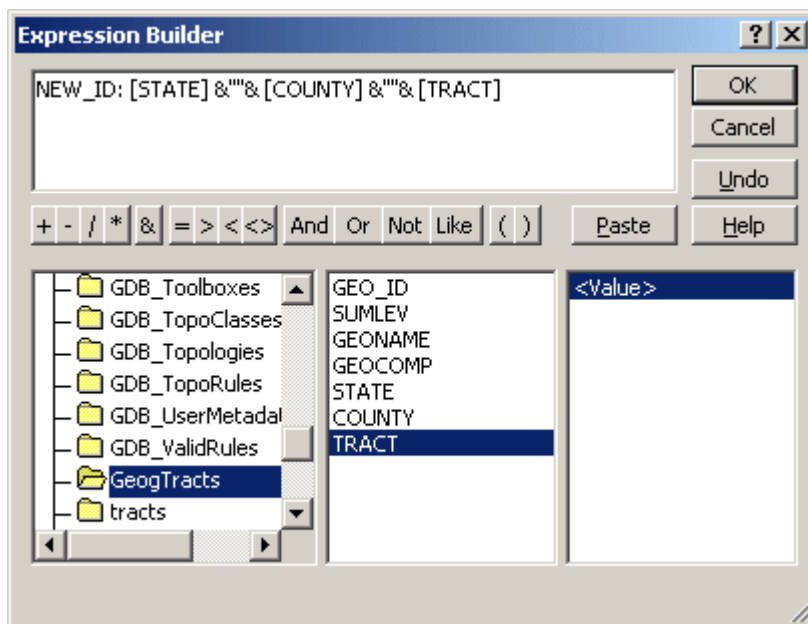
So the next step is to add all of the fields from *GeogTracts* to the query. Select the first field name in the table by left clicking on it, scroll to the bottom of the field list and, while holding down the shift key select the last field name in the list. This should select all of the fields in the list. Then hold down the left mouse button, drag the field names down to the first empty box, and release the button. This will add all of the fields from the table to the query. When we are finished, the window should look like Figure 5:

Figure 5



At this point, our query contains all of the fields that are present in the *GeogTracts* table. The next step is to create the new, concatenated column. Scroll to the end of the list of query fields, to the first available, empty box. Click in the first row (the *Field* row) of the first empty box, so that a blinking cursor appears. Then, go to the tool bar menu and click the Magic Wand. This will launch the expression builder, which consists of four windows: a top window where you type in the expression, a lower-left window that lists all of your tables, a lower-middle window that lists all of the fields within the table that is selected in the lower-left window, and a lower-right value window, which you can ignore. The expression window, with the code you need to concatenate the columns, looks like Figure 6:

Figure 6

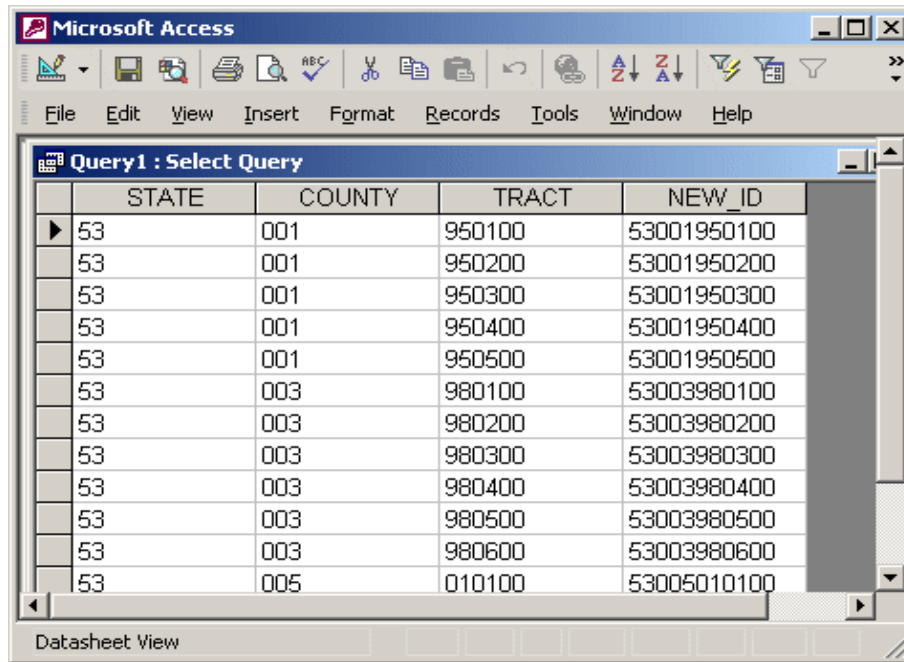


In the first part of the expression, you create a name for the new column that you are going to create. In this case, we create the name `NEW_ID`. We separate the name of the field from the actual expression with a colon. After the colon is a list of all of the fields that we want to combine, with the name of the field in square brackets. You can type in the field names or select them from the list (if you select them from the list, Access will add a placeholder for an expression name and the name of the table - make sure to delete the extra names and characters that it inserts). Then separate each field name with an ampersand, two quotes, and an ampersand `&''&`. This is the syntax that tells Access to concatenate these columns. The final expression should be:

```
NEW_ID: [STATE] &''& [COUNTY] &''& [TRACT]
```

Click OK when you are finished, and the new calculated field will be added to our query in the design view. We can preview the query by hitting the toggle button in the upper-left hand corner of the tool bar (the button has a little table on it). Scroll to the far left, and you will see the new concatenated ID column (see Figure 7).

Figure 7

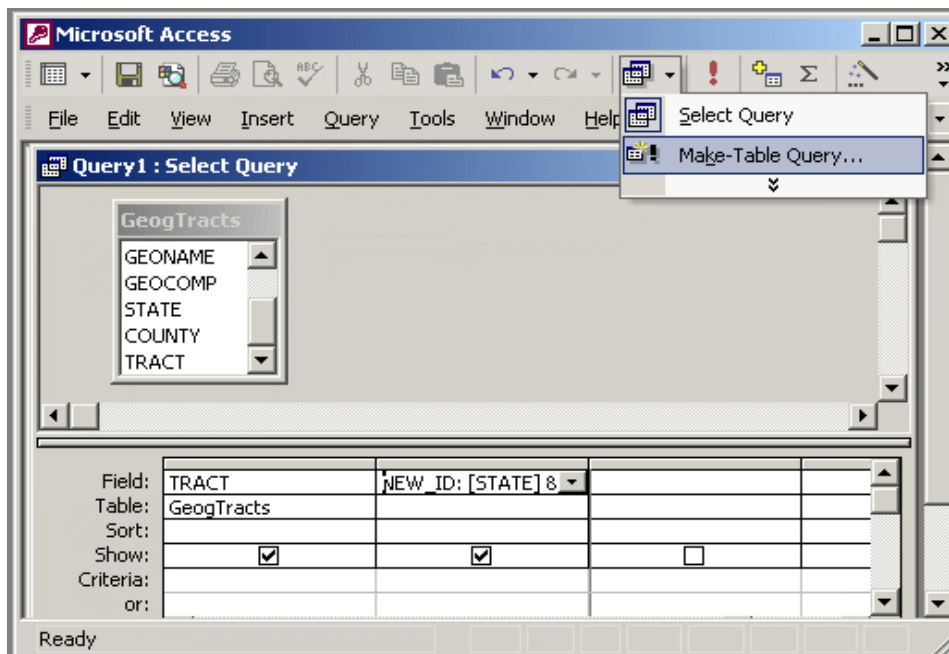


Creating a New Table with the Concatenated Field

At this point, we have a query that has all of our data from the *GeogTracts* table, plus our new concatenated field. But we are not finished yet. At this point, we only have a query. The query itself contains no actual data: it is simply a set of commands that displays data from our table based on our instructions. The next step is to take this query and turn it into a table.

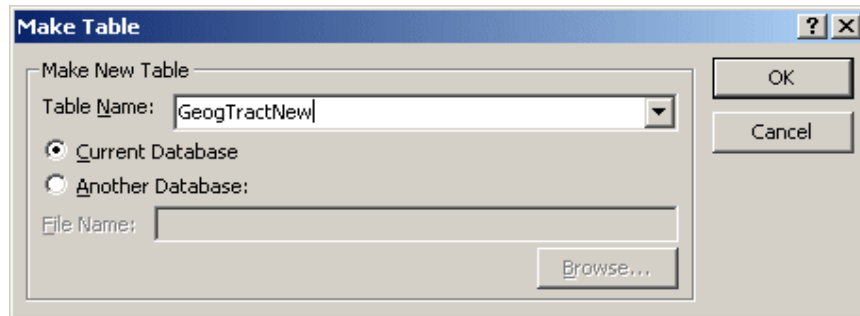
Hit the toggle button in the upper-left hand corner of the tool bar (the button has a little triangle and pencil on it) to return to the query design view. On the tool bar, select the query drop down menu (the button has two tables on it) and select the *Make-Table Query* option (Figure 8).

Figure 8



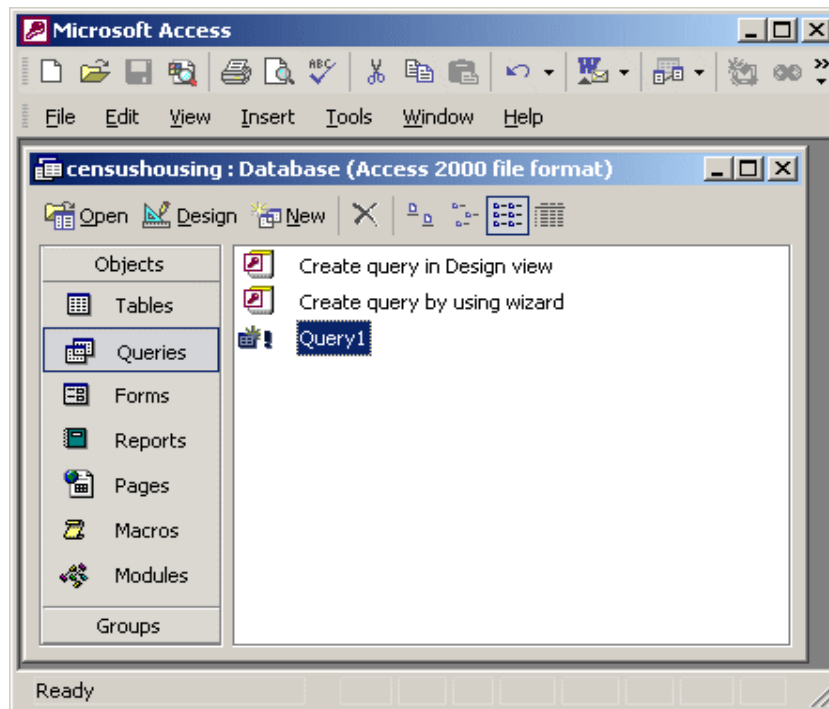
You will be presented with the *Make Table* window, which prompts you to create a name for the table you want to create (Figure 9). Save the table in the current database. Click OK.

Figure 9



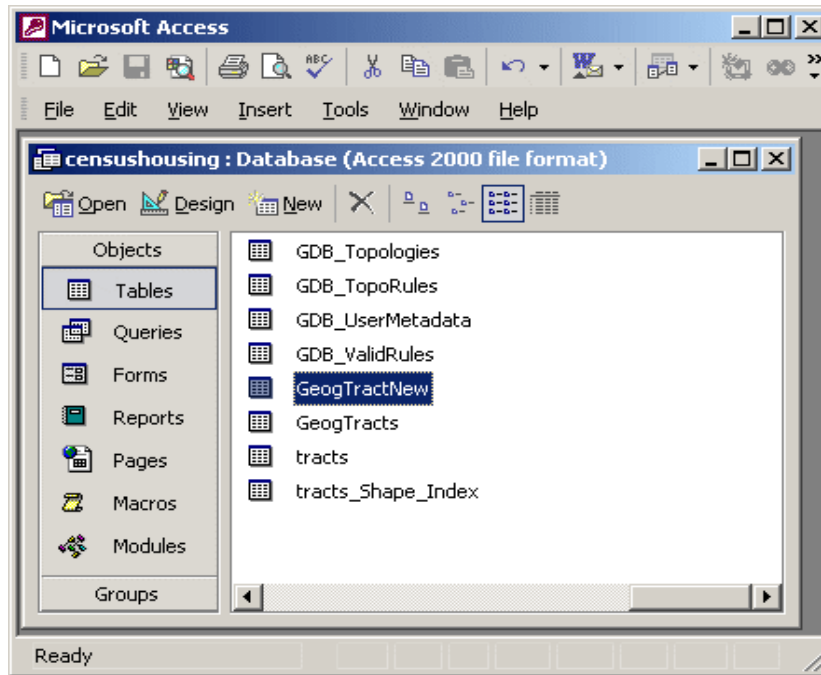
Once you have done this, close the query window. You will be asked whether you would like to save this query. Say yes, and accept the default name, *Query1*. Then you will be back at the query objects window, and you will see one object: *Query1* (Figure 10).

Figure 10



At this stage, we have created a Make-Table query that will: take all of the fields from the *GeogTract* table, add a concatenated ID number field, and will turn this into a new table called *GeogTractNew*. The last step is to actually run this query to make the new table. Double-click on the query to run it. Access will present you with some warning messages that you are about to create a new table. Click OK to get through both of the messages, and Access will run the query. It may take some time, depending on how large the data table is. Once it is finished, click on *Tables* in the objects menu, and scroll through the list of tables. You should see the new table, in this case, *GeogTractNew*, in the list (Figure 11). When you open it, it will look exactly like the query you assembled: it will have all of the fields from the original table, plus the new concatenated ID field.

Figure 11



You now have a brand new table with the new ID column. At this point, you may want to consider deleting the original table, if it is no longer useful. More importantly, you might want to delete the make-table query. If you were to click on the make-table query again to run it, the *GeogTractNew* table would be completely overwritten, and any changes that you made after creating the table would be lost.