

How to Create an Auto-Fill Query Process in MS Access



Mark David Edwards
VVBA Office Solutions, LLC
<http://vbaofficesolutions.com>

Yes it can be done. It's not as simple as just one query, but it's not hard to do. The process is relatively straightforward and reusable, assuming you can change the table structure or create a new table, or don't have to do either (already have the sequence number field). Performance will depend on your system power and the number of records in your table

But first, how did we get in this situation? A lot of times we import Excel reports that are formatted as, well... reports, and not database-friendly tables. These reports can include values as stepped headings with empty cells beneath, where the reader is expected to understand that the heading pertains to the empty cells below the heading. I'm going to assume that there may be instances where a value may appear again later in the table.

What we're going to do is create a table with the values we want to autofill and two fields containing the start and stop record ID values that, when sorted, keeps the records in the proper order for auto-fill. The table will then be used to fill-in the missing values in our data table. The sample table and queries below are an actual working example which you can re-create if you want.

Your Table: First, the data needs to have a sort field that can be used to keep the records in the order that you need to do a fill-down. If you don't already have a field that can be used for this, you can easily create one by adding an autonumber field to the existing table to get the records numbered in ordinal position order.

Whatever you do, you want to be able to sort your table to get the proper autofill record order.

Here's a sample data table with auto-fill target column and sortable column (autonumber field):

Table1	
ValueField	AutoNum
red	1
	2
	3
yellow	4
	5
	6
red	7
	8
	9

Do the following for each field that needs to be autofilled. The explanations below are followed by the query SQL and the results of the query:

Step 1: Create a "MaxAutoNum" summary query that gives the max autonumber value. This will be used as the stop-row value for the last section to be updated.

Query SQL:

```
SELECT Max(Table1.AutoNum) AS MaxAutoNum
FROM Table1;
```

Results:

qry1_MaxAutoNum	
MaxAutoNum	
	9

Step 2: Create a "start-row number" select query that filters out the "empty" rows. This will be the list of "start" rows with the value to be used to fill-in and its "start-row" autonumber. Add the MaxAutoNum query and the MaxAutoNum field.

Query SQL:

```
SELECT Trim([ValueField] & "") AS [Value], Table1.AutoNum, qry1_MaxAutoNum.MaxAutoNum
FROM Table1, qry1_MaxAutoNum
WHERE (((Trim([ValueField] & ""))<>""))
ORDER BY Table1.AutoNum;
```

Results:

qry2_SelectStartRows		
Value	AutoNum	MaxAutoNum
red	1	9
yellow	4	9
red	7	9

Step 3: Use the start-row query in a new summary query, grouping on the Value and start-row fields, and sorting on the start-row field. Then add a 3rd field that will calculate the "stop-row". To calculate the stop-row value, have the field get the Min value of the start-row field that is greater than the start-row value, and subtract 1 (-1) to get the number of the record before the next value record.

Query SQL:

```
SELECT qry2_SelectStartRows.Value, qry2_SelectStartRows.AutoNum AS [Start-Row], Min([qrySelectStartRows_1].[AutoNum])-1 AS [Stop-Row]
FROM qry2_SelectStartRows, qry2_SelectStartRows AS qrySelectStartRows_1
WHERE (((qrySelectStartRows_1.AutoNum)>[qry2_SelectStartRows].[AutoNum]))
GROUP BY qry2_SelectStartRows.Value, qry2_SelectStartRows.AutoNum
ORDER BY qry2_SelectStartRows.AutoNum;
```

Results:

qry3_GetStopRow		
Value	Start-Row	Stop-Row
red	1	3
yellow	4	6

Step 4: Create a make-table query with the select start rows query and add 1 (+1) to the star-row field to get the row directly after the first filled row., then add the stop row query to make a query that makes or appends the records to a table. You'll want to index the Value field.

Query SQL:

```
SELECT qry2_SelectStartRows.Value, [AutoNum]+1 AS [Start-Row],
Val(Nz([qry3_GetStopRow].[Stop-Row],[MaxAutoNum])) AS [Stop-Row]
INTO Table1FillIn
FROM qry2_SelectStartRows LEFT JOIN qry3_GetStopRow ON
(qry2_SelectStartRows.Value = qry3_GetStopRow.Value)
AND (qry2_SelectStartRows.AutoNum = qry3_GetStopRow.[Start-Row]);
```

Results:

qry4_MakeUpdateTable		
Value	Start-Row	Stop-Row
red	2	3
yellow	5	6
red	8	9

Step 5: Finally! The update query: Create an update query with the fill-in table that updates the fields starting with the start-row and stops with the stop-row. (You can alter the start and stop row numbers if you want.)

Query SQL:

```
UPDATE Table1, Table1FillIn
SET Table1.ValueField = [Table1FillIn].[Value]
WHERE (((Table1.AutoNum)>=[Table1FillIn].[start-row]
And (Table1.AutoNum)<=[Table1FillIn].[stop-row]));
```

Data table results:

Table1	
ValueField	AutoNum
red	1
red	2
red	3
yellow	4
yellow	5
yellow	6
red	7
red	8
red	9