Synchronise a table with external data

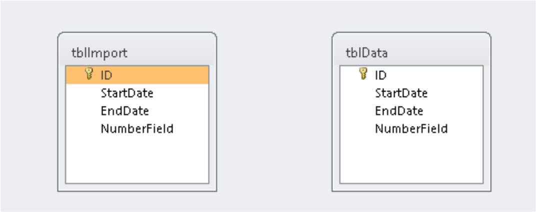
As with most processes in Access, there are several ways of achieving this.
Whichever method is used it is recommended that a backup is made before synchronising with external data in case any problems arise.

The best solution for a particular case will depend on various factors including:

- the number of records to be transferred or modified
- speed and file size

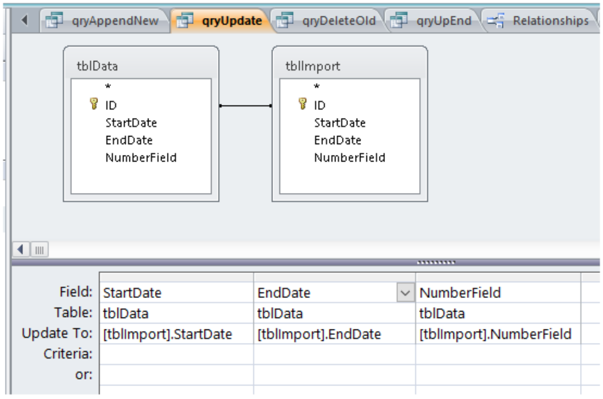
The following code assumes that:

- the records in tblData are being updated from tblImport
- both tables have the same fields and an autonumber primary key field.
- tblImport contains ALL records that should be imported to tblData



1. **Update Existing /Append New / Delete Old**

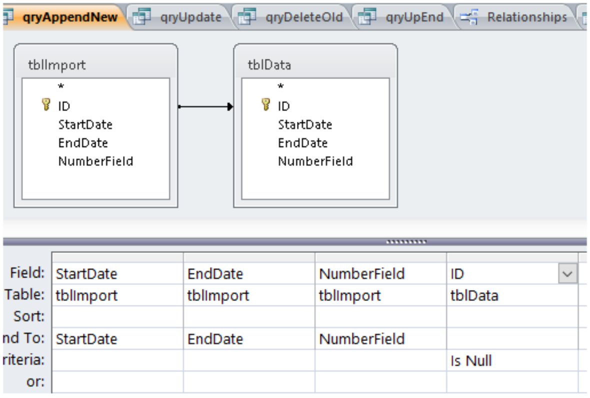
First update existing records



Field:	StartDate	EndDate	NumberField
Table:	tblData	tblData	tblData
Update To:	[tblImport].StartDate	[tblImport].EndDate	[tblImport].NumberField
Criteria:			
or:			

```
UPDATE tblData INNER JOIN tblImport ON tblData.ID = tblImport.ID SET tblData.StartDate =
[tblImport].StartDate, tblData.EndDate = [tblImport].EndDate, tblData.NumberField = [tblImport].NumberField;
```

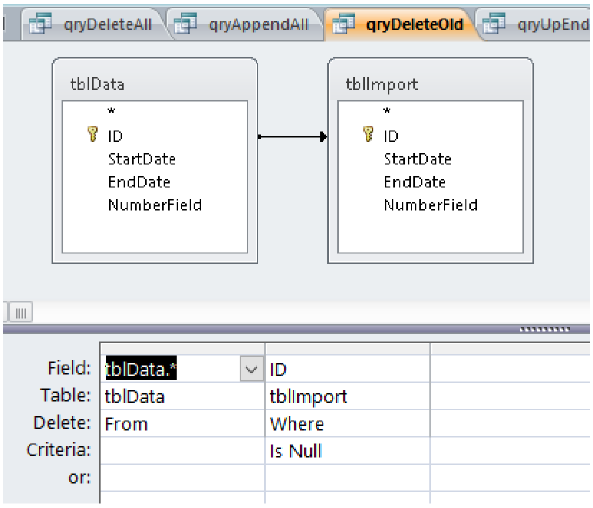
Next append any new records omitting the autonumber ID field



Field:	StartDate	EndDate	NumberField	ID
Table:	tblImport	tblImport	tblImport	tblData
Sort:				
nd To:	StartDate	EndDate	NumberField	
riteria:				Is Null
or:				

```
INSERT INTO tblData ( StartDate, EndDate, NumberField )
SELECT tblImport.StartDate, tblImport.EndDate, tblImport.NumberField
FROM tblImport LEFT JOIN tblData ON tblImport.ID = tblData.ID
WHERE (((tblData.ID) Is Null));
```

Finally delete any old records that aren’t in the import table

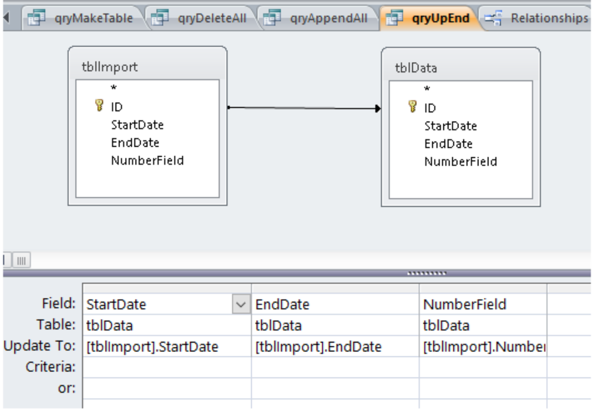


Field:	tblData.*	ID
Table:	tblData	tblImport
Delete:	From	Where
Criteria:		Is Null
or:		

```
DELETE tblData.* FROM tblData LEFT JOIN tblImport ON tblData.ID = tblImport.ID
WHERE (((tblImport.ID) Is Null));
```

2. **Combined Upend (AKA Upsert) / Delete Old**

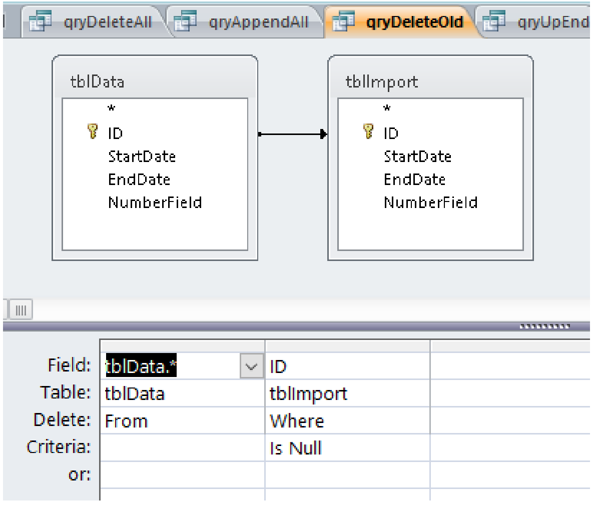
An upend or upsert query can sometimes be used to combine the append and update queries
This is an append query with an outer join
For more details see - see www.mendipdatasystems.co.uk/upend-query/4594428616



Field:	StartDate	EndDate	NumberField
Table:	tblData	tblData	tblData
Update To:	[tblImport].StartDate	[tblImport].EndDate	[tblImport].Numbe
Criteria:			
or:			

```
UPDATE tblData RIGHT JOIN tblImport ON tblData.ID = tblImport.ID SET tblData.StartDate =
[tblImport].StartDate, tblData.EndDate = [tblImport].EndDate, tblData.NumberField = [tblImport].NumberField;
```

Delete any old records that aren’t in the import table (same as above)

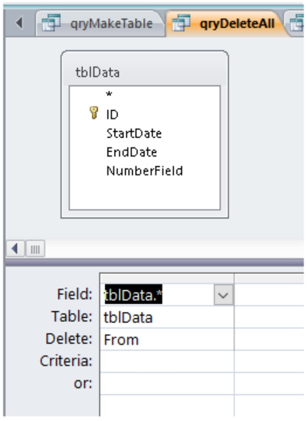


Field:	tblData.*	ID
Table:	tblData	tblImport
Delete:	From	Where
Criteria:		Is Null
or:		

```
DELETE tblData.* FROM tblData LEFT JOIN tblImport ON tblData.ID = tblImport.ID
WHERE (((tblImport.ID) Is Null));
```

3. Delete All / Append All

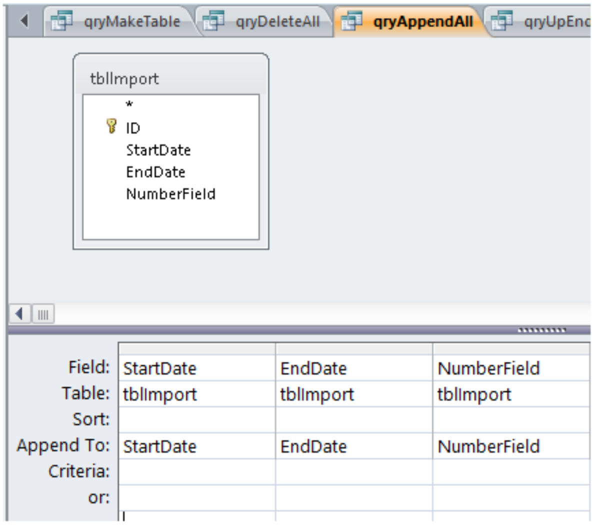
First delete ALL records from tblData



Field:	tblData.*
Table:	tblData
Delete:	From
Criteria:	
or:	

DELETE tblData. FROM tblData;*

Now append all records from the import table

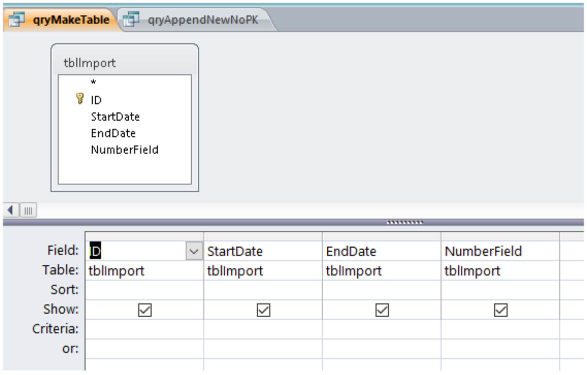


Field:	StartDate	EndDate	NumberField
Table:	tblImport	tblImport	tblImport
Sort:			
Append To:	StartDate	EndDate	NumberField
Criteria:			
or:			

INSERT INTO tblData (StartDate, EndDate, NumberField)
SELECT tblImport.StartDate, tblImport.EndDate, tblImport.NumberField FROM tblImport;

4. Make Table

This will overwrite the existing table



Field:	ID	StartDate	EndDate	NumberField
Table:	tblImport	tblImport	tblImport	tblImport
Sort:				
Show:	☑	☑	☑	☑
Criteria:				
or:				

SELECT tblImport.ID, tblImport.StartDate, tblImport.EndDate, tblImport.NumberField INTO tblData
FROM tblImport;

NOTE: The new table will not have a primary key field unless this is added using code

ALTER TABLE tblData ADD COLUMN ID COUNTER(1, 1) NOT NULL PRIMARY KEY;

SUMMARY

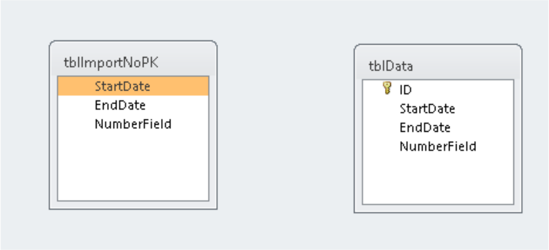
In each of the above cases, file size will increase as new records are being added
Methods 1 and 2 will cause less file 'bloat' as the number of records being added/deleted will be smaller

Methods 3 and 4 will cause more file bloat as all records are being replaced
The fastest method will normally be method 4 followed by method 3
Method 1 will normally be the slowest as Access needs to determine which records to add / delete and run a separate update.
Indexing the fields will make the process slower as the indexes also need to be updated.

Unless you have a large number of records to synchronise the time difference may be negligible

---

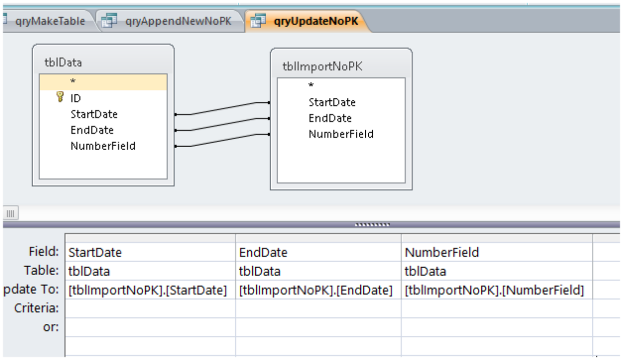
If you are importing from an Excel or CSV file, there is likely to be no primary key field in the import table



This requires some changes to the above code.
More care is needed in checking the outcomes will be what you want before synchronising the data

5. Update Existing /Append New / Delete Old

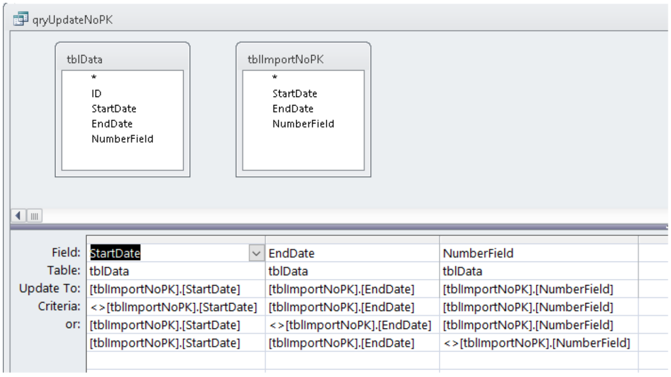
First update existing records.
The following query won't work as it will only update records if they are the same in both tables
This is of course pointless



Field:	StartDate	EndDate	NumberField
Table:	tblData	tblData	tblData
pdate To:	[tblImportNoPK].[StartDate]	[tblImportNoPK].[EndDate]	[tblImportNoPK].[NumberField]
Criteria:			
or:			

DO NOT USE THIS

Instead create a cartesian join query (unlinked tables) and specify conditions to use.
For example, this will update any records where any ONE field has been changed



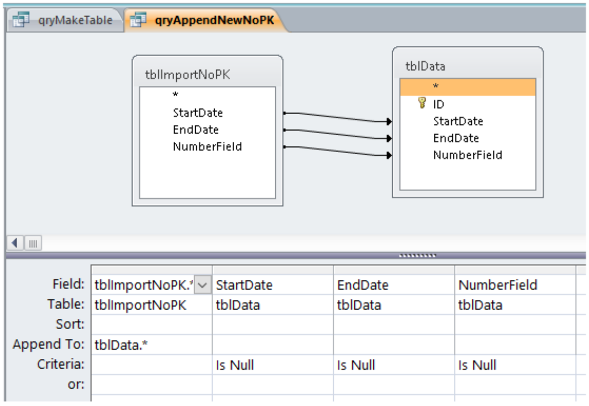
Field:	StartDate	EndDate	NumberField
Table:	tblData	tblData	tblData
Update To:	[tblImportNoPK].[StartDate]	[tblImportNoPK].[EndDate]	[tblImportNoPK].[NumberField]
Criteria:	<>[tblImportNoPK].[StartDate]	[tblImportNoPK].[EndDate]	[tblImportNoPK].[NumberField]
or:	[tblImportNoPK].[StartDate]	<>[tblImportNoPK].[EndDate]	[tblImportNoPK].[NumberField]
	[tblImportNoPK].[StartDate]	[tblImportNoPK].[EndDate]	<>[tblImportNoPK].[NumberField]

As no PK is involved, you should set Unique Records = Yes

```
UPDATE DISTINCTROW tblData, tblImportNoPK SET tblData.StartDate = [tblImportNoPK].[StartDate],
tblData.EndDate = [tblImportNoPK].[EndDate], tblData.NumberField = [tblImportNoPK].[NumberField]
WHERE (((tblData.StartDate)<>tblImportNoPK.StartDate) And ((tblData.EndDate)=tblImportNoPK.EndDate) And
((tblData.NumberField)=tblImportNoPK.NumberField)) Or (((tblData.StartDate)=tblImportNoPK.StartDate) And
((tblData.EndDate)<>tblImportNoPK.EndDate) And ((tblData.NumberField)=tblImportNoPK.NumberField)) Or
(((tblData.StartDate)=tblImportNoPK.StartDate) And ((tblData.EndDate)=tblImportNoPK.EndDate) And
((tblData.NumberField)<>tblImportNoPK.NumberField));
```

NOTE: this will not pick up any records where TWO or more fields have been changed
Additional criteria would be needed to manage those situations.
The query could get very complicated in such cases. If so, this method is best avoided.

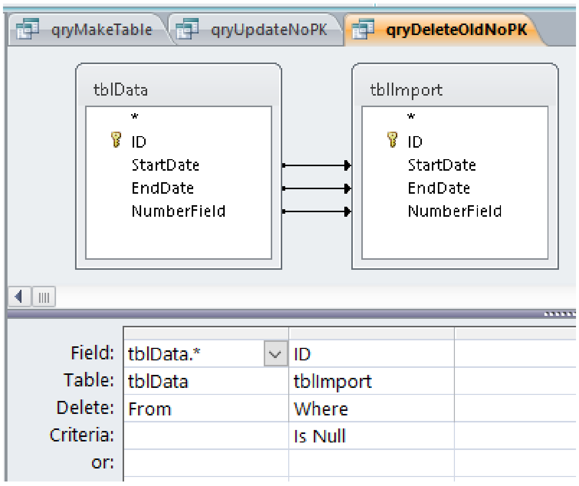
Assuming the method is feasible in your situation, next append all new records.
Again set Unique Records = Yes



Field:	tblImportNoPK.*	StartDate	EndDate	NumberField
Table:	tblImportNoPK	tblData	tblData	tblData
Sort:				
Append To:	tblData.*			
Criteria:		Is Null	Is Null	Is Null
or:				

```
INSERT INTO tblData SELECT DISTINCT tblImportNoPK.*
FROM tblImportNoPK LEFT JOIN tblData ON (tblImportNoPK.NumberField = tblData.NumberField) AND
(tblImportNoPK.EndDate = tblData.EndDate) AND (tblImportNoPK.StartDate = tblData.StartDate)
WHERE (((tblData.StartDate) Is Null) AND ((tblData.EndDate) Is Null) AND ((tblData.NumberField) Is Null));
```

Finally delete any old records that aren't in the import table – again set Unique Records = Yes



Field:	tblData.*	ID	
Table:	tblData	tblImport	
Delete:	From	Where	
Criteria:		Is Null	
or:			

```
DELETE DISTINCTROW tblData.*, tblImport.ID
FROM tblData LEFT JOIN tblImport ON (tblData.StartDate = tblImport.StartDate) AND (tblData.EndDate =
tblImport.EndDate) AND (tblData.NumberField = tblImport.NumberField)
WHERE (((tblImport.ID) Is Null));
```

6. Combined Upend (AKA Upsert) / Delete Old

This method cannot be used as there is no PK field in the import table

7. **Delete All / Append All**

   This is identical to method 3 above

   First delete ALL records

```
DELETE tblData.* FROM tblData;
```

   Now append all records from the import table

```
INSERT INTO tblData ( StartDate, EndDate, NumberField )
SELECT tblImportNoPK.StartDate, tblImportNoPK.EndDate, tblImportNoPK.NumberField FROM tblImportNoPK;
```

8. **Make Table**

   As there is no PK field in the import table, this method can only be used if the PK field is added separately

SUMMARY

In cases where there is no PK field in the import table, your choices are more limited.
Take great care to ensure the results are what you want before synchronising data

Use a SELECT query to check the records before running a DELETE, APPEND or UPDATE query

---

NOTE:

a) Compacting will recover much of the space added after deleting records

b) Repeatedly creating / overwriting tables can cause instability & in some cases may lead to corruption

c) The example database **SyncDataExample.accdb** contains all the tables / queries used in this article

Colin Riddington *Mendip Data Systems* *09/01/2019*