

Re: Hard to explain – multiple update fields query...

Source:

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.queries/2008-07/msg00773.html>

- *From:* David Mulholland <DavidMulholland@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Thu, 17 Jul 2008 05:22:01 -0700
-

Thanks guys...you've all got some great ideas. I'll be tackling this today. I'll let you know how it goes. Thanks again.

"John Spencer" wrote:

For some reason my response got truncated. You would use an update query to get the data back into the AwardsTable

```
UPDATE AwardsTable as A INNER JOIN TempTable as T
ON A.ID = T.ID
SET A.Award1 = [T].[Award]
, A.Award1Date = [T].[AwardDate]
, A.NumberAwarded1 = [T].[NumberAwarded]
WHERE T.AwardPrecedence = 1
```

Repeat the above for each precedence level changing the set of fields you update. For instance, for the 17th set you would see something like:

```
UPDATE AwardsTable as A INNER JOIN TempTable as T
ON A.ID = T.ID
SET A.Award17 = [T].[Award]
, A.Award17Date = [T].[AwardDate]
, A.NumberAwarded17 = [T].[NumberAwarded]
WHERE T.AwardPrecedence = 17
```

If you were paranoid and the structure allowed it, you might just add new records with all the data and then when you were satisfied that all went correctly, delete the old one.

Another option, would be to build a parallel structure of the AwardsTable in your Access database and do all the work there. Once you were confident that the process worked you could either update all 40 sets at once from the Access table or import the entire set of records at once.

Of course this entire solution depends on you being able to lock the

Re: Hard to explain – multiple update fields query...

AwardsTable for data entry for the entire period you need to do the work.

```
'=====
John Spencer
Access MVP 2002–2005, 2007–2008
The Hilltop Institute
University of Maryland Baltimore County
'=====
```

John Spencer wrote:

Ok, then try restructuring the data into a table you create in your Access database. Data structure – field and tables names – are all generic guesses.

Fields:
ID
Award
AwardDate
NumberAwarded
AwardPrecedence

You will need 40 queries to populate the table from your current table.

```
INSERT INTO TempTable (ID, Award, AwardDate,NumberAwarded,
Precedence)
SELECT ID, Award1
, Award1Date, NumberAwarded1
, AwardsPrecedenceTable.Precedence
FROM [AWARDSTable] Left JOIN AwardsPrecedenceTable
ON AwardsTable.Award = AwardsPrecedenceTable.Award
WHERE Award1 is Not Null
```

Now you have the data normalized and can shove it back into your original table. Perhaps cleaning out the 40 sets of fields first using an update query to set all the fields in the set to null and then using an update query to populate the sets.

You could do one set at a time if you wish. Clear out set 1 and populate it, clear out set 2 and populate it.

STEP 1: BACKUP your data before attempting the following.
STEP 2: BACKUP your data before attempting the following.

Without a backup you cannot restore the data if this does not work the way you expect.

Re: Hard to explain – multiple update fields query...