

## Re: Editing Union results

**Source:** <http://www.tech-archive.net/Archive/Access/microsoft.public.access.queries/2004-12/1951.html>

---

**From:** corey lawson (*corey.lawson\_at\_worldnetdotattdotnet*)

**Date:** 12/18/04

Date: Sat, 18 Dec 2004 11:08:06 GMT

John Vinson <jvinson@STOP\_SPAM.WysardOfInfo.com> wrote in news:p189r05ikgbkpp3qp9a1iv8oga53t0ni8p@4ax.com:

> *On Mon, 6 Dec 2004 05:49:03 -0800, ptb476*  
> *<ptb476@discussions.microsoft.com> wrote:*  
>  
>> *I am trying to create a form driven by a Union query where the users*  
>> *can edit the results.*  
>  
> *You can't. UNION queries are never updateable and cannot be made*  
> *updateable.*

Actually, they can be made editable in some backend RDBMS, like Oracle 8i+, SQL Server 7.0+, etc., by using "INSTEAD OF" triggers on views made up of "uneditable" queries like union queries.

The INSTEAD OF trigger "catches" the insert, update and/or delete events and handles things.

It's either that, or write some sort of code (Stored Proc, VBA function) that does this for you, but this is MUCH harder to pull off for Access, because if a form's underlying dataset is read-only, your form's bound controls will be read-only. So you end up having to write event handlers to scrape-and-paste into unbound form controls, and then somehow turning the data from the form back into another function back into the database. Icky stuff worthy of figuring out a better way to do it in your database design.

Chances are, your design is a common database design mistake. Instead of having different tables with common information, you have one table with the common information, and separate tables with the non-common information. Something like this:

```
tblPerson:
  personID
  personTypeCode
  lastName
  firstName
```

```
tblEmployee  
  personID  
  employeeID
```

etc...

And, if you don't code the "person" code in a VBA Class, then you end up having to do icky repetitive code because Access tables don't expose anything approaching the functionality of triggers, but if your back-end database is Oracle, SQL Server, MSDE, etc., then write triggers on the table to create "child" records for you, which in this case might be like (almost T-SQL):

```
set @newPersonID = select PersonID from inserted
```

```
if Inserted.personTypeCode = 'EM'  
  insert into Employee (personid) values(@newPersonID)
```

...and so on.

Probably the best you could do in the read-only form case is to have it pop up various editable forms on the chunks of data these can all update. Of course,