

# Re: Concatenate and Null Values -- Features

---

*Source:*

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.reports/2007-01/msg00460.html>

---

- *From:* Duane Hookom <[duanehookom@xxxxxxxxxxxxxxxxxxxxx](mailto:duanehookom@xxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Fri, 12 Jan 2007 11:45:01 -0800
- 

Any solution where you need to modify fields, tables, forms, reports, queries, etc to add or remove tests is not normalized. I created an application that schedules and records results from various tests (actually sampling for bacteria or other "bugs"). My largest table has 14 fields which probably should be normalized more. I (actually users) can add more test types at any point without changing anything other than data.

Without going in to all the details, some of my tables are (there are actually 19 tables):

tblTests (1 record per test type)

=====

tstTstID autonumber primary key  
tstTitle values like Coliform, bacteria,...  
this would be "low width", "High width",...  
if you want to add another test, add a record  
tstStatus  
tstComments

tblTestRanges

=====

(this table describes various levels of results for each test. We also have this divided by site/area since finding a bacteria in the parking lot isn't as critical as in the plant)

tblSamples

=====

samSamID autonumber primary key  
samSitID links to tblSites.sitSitID (where)  
you might link to an item tested  
samDate date of sample  
samEmpID who took the sample  
samComments

tblSampleTests

=====

smtSmTID autonumber primary key  
smtTstID link to tblTests.tstTstID

smtResult numeric result of test  
smtComments  
smtTestDate Date Tested  
smtEmpID test performed by  
--

Duane Hookom  
Microsoft Access MVP

"Hoopster" wrote:

To all who are trying to help:

I hate to be a pest but I have been reading ACC: Database Normalization Basics and trying to learn everything I can. As a matter of fact, I have printed everything pertaining to Database Normalization that I can find. I really want to do this right.

Comparing what I am reading with my table structure, I m not seeing what looks to be a lot I can change. Don t I have to keep all of my Audit Data together in one Table? The Table that the above structure was for, was the table holding all my data. I can see where I could change some things like everything down to and including Item Number could be kept in a separate Data Table and linked to this one using the Audit Number (key field). Also all the cells like Low Header, High Header, Low Fly, High Fly, ect. could be left out of this table and access them by linking with my Tolerance table by the Flag Style. The reason I did it this way was if the Tolerance was changed, the data record would show what Tolerance the Audit was based against. As far as the others like Header Stitch #1 thru #5 and Fly Hem #1 thru #5, ect., these are all legit, each one representing a separate Audit. Header Stitch #1 would be for Flag Audit 1, Header Stitch #2 would be for Flag Audit 2, ect. 5 Audits per record.

Below are the other Tables I have pertaining to our Flag Audits and there structure. I wanted to ask this question before I started changing and messing everything up that I have done to date.

Audit Item Number List

=====  
Record ID 4 4  
Flag Type 10 75  
Item Number 10 50

Audit Vendor List

=====  
Record ID 4 4  
Vendor 10 50

Department List

=====  
Record ID 4 4  
Department 10 50

Re: Concatenate and Null Values -- Features

Department Name 10 50  
Department Supervisor 10 50

Department Titles

=====

Record ID 4 4  
Department Title 10 50

Audit Tolerance Table

=====

ID 4 4  
Flag Style 10 255  
Header Stitch Low 6 4  
Header Stitch High 6 4  
Fly Hem Low 6 4  
Fly Hem High 6 4  
Side Hem Low 6 4  
Side Hem High 6 4  
Side Hem Width Low 6 4  
Side Hem Width High 6 4  
Low Width 6 4  
High Width 6 4  
Low Length 6 4  
High Length 6 4  
Low Grommet 6 4  
High Grommet 6 4  
Label Type 10 50  
Low Label 6 4  
High Label 6 4  
Low Header Balance 6 4  
High Header Balance 6 4  
Tail 10 50  
Finish Fold Width Low 6 4  
Finish Fold Width High 6 4  
Finish Fold Length Low 6 4  
Finish Fold Length High 6 4  
Back Tack Low 6 4  
Back Tack High 6 4  
SPI Red Low 6 4  
SPI Red High 6 4  
SPI White Low 6 4  
SPI White High 6 4  
SPI Blue Low 6 4  
SPI Blue High 6 4  
Self Header Low 6 4  
Self Header High 6 4  
Low FOS 6 4  
High FOS 6 4  
Low Diagonal 6 4  
High Diagonal 6 4  
Diagonal Tolerance 6 4

## Re: Concatenate and Null Values -- Features

"BruceM" wrote:

I have to say it's hard for me to see how a sequence number such as I have described or a sequential invoice number (00001, 00002, etc.) is meaningful except that it is the real-world reference number that people will use to identify the form, invoice, or whatever. In the end it just represents a chronological sequence of records.

Here's something I'm wondering. I can store 01, 02, etc. and combine that with the year part of a date field, but I cannot use a unique index for the field because it starts over with 01 next year. If it is indexed (allow duplicates), I don't see how the order will be meaningful, so I suppose the best thing would be not to bother with an index on that field at all. I could sort by date, since that is the order in which I wish the records to appear (filtered in some cases, but in chronological order), but I suppose that would happen in a query, or in a form's OrderBy property, or in a report's Sorting and Grouping, rather than as an index.

I should probably mention that this database is unlikely to generate more than 30 records per year, so inefficient indexing and sorting will probably make little discernible difference. However, I'd like to develop good habits now, so I try to determine what would be best with millions of records. I wonder if storing 07-01 (probably without the dash, which I would add just for display) would allow more efficient ordering of the records.

I think I know what you're saying about learning by teaching. While answering questions in these forums is not quite the same as teaching, sometimes I see a problem that I know I would like to solve some day, so I go ahead and experiment with it. Sometimes I find the question has already been answered by the time I come up with a solution, but that's OK, because it usually either confirms what I have done or shows me a better way to do it.

"strive4peace" <strive4peace2006@xxxxxxxx> wrote in message [news:OCDycPdNHHA.5104@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:OCDycPdNHHA.5104@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx)

Hi Bruce,

IMHO, every table should have an autonumber primary key (even if you do not need it for a relationship) and it is not a good idea to use meaningful data for relationships.

On the field with 07-01, 07-02, etc, you could set a Unique index so duplicates would not be allowed. If you HAVE the date this comes from, there is really no reason to store it because it can always be calculated.

## Re: Concatenate and Null Values -- Features

"embarrassed by some of my early attempts"

I know how you feel! Every day, I learn more about Access -- each application is better than the last ...

I learned Access by teaching it; I had been programming with DOS-based databases for about 10 years when I was approached to teach Access running under Windows 3.0 <g> ... they just asked if I could do it, not if I had ever used it ;) So I crammed like crazy and no one ever knew that the teacher was as new as the students! I researched every question that anybody had, and was usually making handouts for the afternoon during my lunch break. Teaching is a great way to learn.

Warm Regards,  
Crystal

\*

(: have an awesome day :)

\*

MVP Access  
Remote Programming and Training  
strive4peace2006 at yahoo.com

\*

BruceM wrote:

I've only been doing this for a few years (and not full-time at that), and am already embarrassed by some of my early attempts. I am aware of Michael Hernandez' Database Design for Mere Mortals, which I will be acquiring as soon as I get organized enough to order it. The light has come on for me, but I'm still squinting a lot. I'm also trying to decide how much improvising I can do. For instance, I have a situation in which a number increments 07-01, 07-02, etc., then starts over with 08-01 etc. next year. It is a unique number, so I could make it the PK, but I would be storing part of the date, which is already

Re: Concatenate and Null Values -- Features

in another field. I could either add a PK field, combine two fields into a PK (although I'm not sure what the other field would be), or just go ahead and store 07-01, etc. In the first case I am creating a new field in the parent table (autonumber, maybe); in the second I am creating a new field in the child table, since a two-field PK needs a two-field FK (as I understand it); in the third case I have one PK and one FK, but arguably some redundancy in the date. So I ask myself if storing part of the date like that is less redundant than adding a field just so I can avoid storing part of the date. I've about decided to store 07-01, but my point is that I am thinking about this in a way that would not have been possible a year ago.

"Tom Wickerath" <AOS168b AT comcast DOT net> wrote in message [news:D088C525-31BB-4DB7-AF4A-167A33ED48B5@xxxxxxxxxxxxxxxxxxxx](mailto:news:D088C525-31BB-4DB7-AF4A-167A33ED48B5@xxxxxxxxxxxxxxxxxxxx)

Hi Bruce,

For what it's worth, I floundered like a fish out of water for a few years with Access. I had been attempting to work with Access since version 2 was released. The light finally came on for me one evening back in '96 (or perhaps '97) when I was at a meeting of the Pacific NW Access Developer's Group. The speaker was Michael Hernandez, that author of the paper I cited in my previous post. Several of the examples of problems he showed were the

Re: Concatenate and Null Values -- Features

exact types of problems I  
had been encountering. I  
made an effort to  
start  
studying database design  
after that, even though the  
subject can be a  
bit  
dry. The effort really payed  
off.

Tom Wickerath  
Microsoft Access MVP

[http://www.access.qbuilt.com/html/expert\\_contributors.html](http://www.access.qbuilt.com/html/expert_contributors.html)  
<http://www.access.qbuilt.com/html/search.html>

---

"BruceM" wrote:

Hoopster,  
this is a lot  
of stuff  
being tossed  
your way,  
but it is  
really  
good stuff.  
The people  
who have  
added to  
this thread  
since you  
posted  
the  
table  
structure are  
very skilled  
and  
knowledgeable  
(excepting  
this  
posting;  
I can claim  
only that I  
am making  
decent  
progress for  
a relative

## Re: Concatenate and Null Values -- Features

beginner,  
but I know  
good advice  
when I see  
it). I urge  
you not to  
be daunted  
by  
all  
of the new  
information.  
Once you  
start to get  
the hang of  
relational  
databases  
you will  
have a  
powerful  
tool at your  
disposal.