

Re: Mixed up with Relationships..help!

Source:

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.gettingstarted/2005-01/1330.html>

From: Chris2 (rainofsteel.NOTVALID_at_GETRIDOF.luminousrain.com)

Date: 01/26/05

Date: Tue, 25 Jan 2005 21:03:57 -0800

"KrazyRed" <KrazyRed@discussions.microsoft.com> wrote in message news:19FA1AD6-D193-4EAD-82A2-3C64E5C33D4F@microsoft.com...

> *This looks amazing, but i am going to demonstrate how thick I am.*

>

> *The Structure of the database has 1 questionnaire which has gone to 10*

> *different schools. So 10 schools have had all the same questions.*

Also, the

> *sudents themselves are not identified, so the studentID will only be the*

> *unique key to 1 questionnaire. there are about 150 questionnaires completed*

> *for each school.*

Ah, and now I know far more than I did before. :)

The tables I tossed out were based on some assumptions on my part, which turned out to be wrong.

It has the advantage of reusability. The database you're describing above can only be used for that one questionnaire, mine can be used for unlimited numbers of questionnaires, and supports question and answer reuse between questionnaires, etc. People building new questionnaires could have access to a "questionnaire" builder Form which let them use questions on file to help them in the build process.

Take the rest of my comments in this post in the correct context of "I designed something with more capability than you wanted."

>

> *So I don't think we need a sponsors table, and i am slightly puzzled by*

> *reference to CONSTRAINT – what is this?*

The CONSTRAINT statement establishes a Primary Key or a Foreign Key, and is a part of "Referential Integrity" (in MS Access; too bad Access

doesn't support CHECK . . .).

If you have Access 2K, open help, and type in "referential" in the search box, and you should get an entry for "What is referential integrity." (Other versions of Access may list it differently.)

In the table design page, you do the equivalent when you select a row and click on the "key" button on the toolbar.

Access gets its user-friendliness by hiding most SQL (like the CREATE TABLE statement, etc.) from you. Access also hampers learning by hiding this information from you. It's a catch-22.

But when it comes to examples, do I write two pages of how to mouse-click the way to a new table, or do I simply write the 3-8 line CREATE TABLE statement? (Which you are free to cut and paste and use; although it's to your benefit to study and learn it.)

>
> *Am i right in thinking that there should be a table for questions which hold*
> *all the questions for the questionnaire.*

You can, and it's probably the best way.

>
> *and then a table for answers which has the questionID as a foreign key; what*
> *does REFERENCES mean?*

CONSTRAINT <primary-key-name> PRIMARY KEY (<column-name>, . . .)

--The above establishes a "Primary Key". MS Access does this by creating a unique not-null index on the column (or combination of columns). "A Primary Key is the column or combination of columns whose unique value *identifies* the row itself." Technically, in a "real" "relational" database, no Table would ever be without a Primary Key, although almost all actual relational database products allow you to do it.

CONSTRAINT <foreign-key-name> FOREIGN KEY (<column-name>, . . .)
REFERENCES <other-table>
(<column-from-other-table>, . . .)

--The above establishes a "Foreign Key". MS Access does this by creating a unique/non-unique (designer choice) not-null index on the column (or combination of columns). When this happens, every time you try to INSERT to this table, Access make sure that the value about to go into the Foreign Key column is found somewhere in the "REFERENCES" table and column. Every time you try to UPDATE or DELETE, Access does the same thing.

—In the example table Answers, there is a Foreign Key to Questions. If you try to INSERT a row into Answers that has a QuestionID of 2041, and Access looks at Questions, and finds no value 2041 in the QuestionID column there, it stops the INSERT (same for UPDATE and DELETE). Trying to do any of these three things when you shouldn't usually produces some sort of "key violation".

—Understanding Primary and Foreign Keys, what they mean, what they're for, and where they're found, helps you to understand the source of some error messages you will see. If you work solely with the Access UI, all this "key" stuff is hidden behind the "Relationships" window. But if you run an INSERT that says, "Could not INSERT 21 records because of key violations," and you don't know what "keys" are, there's going to be difficulty in diagnosing the problem.

>
> *Then we have a table for questionnaires, which i think i understand, and a*
> *table for questionnaire questions which has me more puzzled. I thought we had*
> *questions already, are these different questions?*

It can be done either way.

In my example:

I created a Table to hold "Questions" (things that could be asked).

I created a Table to hold "Answers" (valid responses to Questions).

I created Questionnaires to hold Questionnaires.

I created QuestionnaireQuestions to hold a "group" of Questions for a particular Questionnaire.

I created QuestionnaireAnswers to hold a student's responses to the Questions on a Questionnaire.

You require less than that, though, as you outlined above.

>
> *And then we have questionnaire answers which i thought we already had answers.*

Answers holds *valid* and true answers provided by the people making the Questionnaire.

Questionnaire = "Cafeteria Food"

Question = "Do you like cafeteria chocolate pudding?" (QuestionID = 45)

QuestionnaireQuestions: QuestionnaireQuestionID (2005), "Cafeteria Food", QuestionID (45)

Valid answers in Answers: Strongly Dislike, Dislike, Take-it/Leave-it, Like, Strongly Like, Allergic, Never Tried.

(Appearing in drop-down combo box).

QuestionnaireAnswers: QuestionnaireAnswersID (51349), StudentID, 2005,
"Dislike"

Follow the chain of ID numbers.

>
> *Are you suggesting that i have a separate table for Questions*
> *and Answers, and questionnaire questions and answers. I would like*
to try
> *and keep this as simple as possible. If u don't mind bearing with*
me, and
> *explaining the terms u have used I would really appreciate it.*

You can make it as simple as you wish. Also bearing in mind that
you will only use it once, and when someone asks you to do it a second
time, a third time, and a fourth time, each with new, empty copies of
the previous, and then someone comes along and asks you to compare the
information, track the response levels between them, etc.

Sorry . . . I tend to think in terms like that.

>
> *You can see why I am confused I hope.*
> *"Chris2" wrote:*
>
>>
>> *"KrazyRed" <KrazyRed@discussions.microsoft.com> wrote in message*
>> *news:95DA600E-7DE2-4FF7-8CA2-5286EB91B609@microsoft.com...*
>>> *Ok, I have managed to get myself completely confused.*
>>>
>>> *I have a table, called tblstudent, which has a series of*
questions
>> *in it, as*
>>> *in data entry for a questionnaire.*
>>
>> *The name tblstudent implies that the table is about students,*
not
>> *about questions.*
>>
>> *That table sounds like it should be called QuestionnaireAnswers.*
>>
>>>
>>> *I have read some books about relationships and the trouble i*
have is
>> *that*
>>> *several of the questions have the same choice of answer, for*
>> *example,*
>>> *tblagreetype. ie. strongly agree, slightly agree, slightly*
disagree

```
> > etc. When
> > > i try to create the lookup table to link to this table, the
> > relationships
> > > window looks all over the place.
> >
> > The Tables would go like this:
> >
> > CREATE TABLE Students
> > (StudentID AUTOINCREMENT
> > ,NameFirst TEXT(72)
> > ,NameMiddle TEXT(72)
> > ,NameLast TEXT(72)
> > ,CONSTRAINT pk_StudentID PRIMARY KEY (StudentID)
> > )
> >
> > CREATE TABLE Faculty
> > (FacultyID AUTOINCREMENT
> > ,NameFirst TEXT(72)
> > ,NameMiddle TEXT(72)
> > ,NameLast TEXT(72)
> > ,CONSTRAINT pk_Faculty PRIMARY KEY (StudentID)
> > )
> >
> > CREATE TABLE Questions
> > (QuestionID AUTOINCREMENT
> > ,Question TEXT(255) NOT NULL
> > ,CONSTRAINT pk_Questions PRIMARY KEY (QuestionID)
> > )
> >
> > CREATE TABLE Answers
> > (AnswerID AUTOINCREMENT
> > ,QuestionID LONG NOT NULL
> > ,Answer TEXT(255) NOT NULL
> > ,CONSTRAINT pk_Answers PRIMARY KEY (AnswerID)
> > ,CONSTRAINT fk_Answers_Questions FOREIGN KEY (QuestionID)
> > REFERENCES Questions (QuestionID)
> > )
> >
> > CREATE TABLE Questionnaires
> > (QuestionnaireID AUTOINCREMENT
> > ,QuestionnaireName TEXT(72) NOT NULL
> > ,CONSTRAINT pk_Questionnaires PRIMARY KEY (QuestionnaireID)
> > )
> >
> > CREATE TABLE QuestionnaireSponsors
> > (QuestionnaireSponsorID AUTOINCREMENT
> > ,QuestionnaireID LONG NOT NULL
> > ,FacultyID LONG NOT NULL
> > ,CONSTRAINT pk_QuestionnaireSponsors PRIMARY KEY (StudentID)
> > ,CONSTRAINT fk_QuestionnaireSponsors_QuestionnaireID FOREIGN KEY
> > (QuestionnaireID)
```

> > REFERENCES

Faculty

> > (QuestionnaireID)

> >)

> >

> > CREATE TABLE QuestionnaireQuestions

> > (QuestionnaireQuestionsID AUTOINCREMENT

> > ,QuestionnaireID LONG NOT NULL

> > ,QuestionID LONG NOT NULL

> > ,CONSTRAINT pk_QuestionnaireQuestions PRIMARY KEY (QuestionnaireID)

> > ,CONSTRAINT fk_QuestionnaireQuestions_Questionnaires FOREIGN KEY

> > (QuestionnaireID)

> > REFERENCES

> > Questionnaires (QuestionnaireID)

> > ,CONSTRAINT fk_QuestionnaireQuestions_Questions FOREIGN KEY

> > (QuestionID)

> > REFERENCES

Questions

> > (QuestionID)

> >)

> >

> > CREATE TABLE QuestionnaireAnswers

> > (QuestionnaireAnswersID AUTOINCREMENT

> > ,StudentID LONG NOT NULL

> > ,QuestionnaireQuestionsID LONG NOT NULL

> > ,StudentsAnswer TEXT(255) NOT NULL

> > ,CONSTRAINT pk_QuestionnaireAnswers PRIMARY KEY (QuestionnaireID)

> > ,CONSTRAINT fk_QuestionnaireAnswers_Students FOREIGN KEY

(StudentID)

> > REFERENCES Students

> > (StudentID)

> > ,CONSTRAINT fk_QuestionnaireAnswers_QuestionnaireQuestionsID

FOREIGN

> > KEY (QuestionnaireQuestionsID)

> >

REFERENCES

> > QuestionnaireQuestions (QuestionnaireQuestionsID)

> >)

> >

> >

> > There, that looks right (bear in mind I wrote that in about 30

> > minutes).

> >

> >

> >

> >

> >