

Using Computers

Database Module

Task 11

Linking tables in relationships

One of the golden rules of database design is that data should exist only in one place. For instance, if a person who orders various items from a bookshop on a regular basis, the sales person would have to enter the persons details every time he orders a book. In the case of the customer moving addresses, the sales person would have to go into the database and find all the persons sales transactions and amend the delivery address. This can be time consuming and not all records may be found.

To overcome this problem, a good database design means that information should be entered only once. Separate tables need to be created; one for customer and another for products sold in the shop. Once these tables are linked together (you will notice a plus sign(+)) in the left most column. Clicking on this sign displays the relevant details from the other linked table.

In the old style, so-called flat file database like in task 1, the database was just a single table of records. In relational databases such as those you can create in Access, however the single database can contain lots of different tables that are linked with each other to eliminate duplications of data entry. Moreover, because the information is entered only once and is held in one place, it is easier to update.

Explore relationship types

Relationship type	Description	Example
One-to-One relationship	A record in the first table can have no more than one matching record in the second table. Typically, the two tables have the same primary key. One-to-one relationships are rare in relational databases.	The storage of sensitive or private information, such as user passwords, salary information, exam grade or disciplinary records, in a separate table from the one where the main details are held. Another reason that a database designer might want to keep information separate is if the content of one table changes more frequently than the data contained in the other, or if the data in different tables need to be updated by different departments.

Relationship type	Description	Example
One-to-many relationships	<p>A record in the first table can have many matching records in the second table, but a record in the second table can have only one matching record in the first table.</p> <p>This is a very common type of relationship.</p>	<p>The first table could list products and the second could list sales transactions.</p> <p>A single produce record could link to many sales transaction records in the second table. However, each transaction record can include a single product.</p>

Access imposes two restrictions on the fields that can be linked in one-to-one and one-to-many relationships.

- Only fields that are either primary keys or have non-duplicated indexes may be linked.
- Only fields of the same data type may be linked i.e. text fields can be link to text fields, number fields to other number fields. In addition, when the linked fields are number fields, they must have the same field size property setting. One exception is the AutoNumber field type. It can be link to a field of a different type in another table.

Lets see what happens with different relationships!

Create yourself a database with the following information. Name the database Bookworm.

ISBN	Pages	Title	Copyright Year	Cost	Purchased Date	Comments
010-9	302	Emma	1816	£4.99	30/11/2001	
011-7	345	Jane Ayre	1847	£4.99	13/09/1998	
016-8	412	Great Expectations	1861	£4.99	13/09/1998	
043-9	280	Cowboys and Indians	1991	£8.99	07/07/1999	
066-4	287	Mansfield Park	1814	£8.99	03/04/1999	
069-3	344	Little Women	1868	£6.99	24/02/1998	
242-4	289	The Shipping News	1993	£6.20	03/04/1999	
274-5	298	Harry Potter and the Philosopher's Stone	1997	£4.99	24/02/1998	
389-X	370	Love in the Time of Cholera	1985	£5.95	24/02/1998	
408-2	432	The History of the Kelly Gang	2000	£13.60	23/10/2000	

ISBN	Pages	Title	Copyright Year	Cost	Purchased Date	Comments
414-2	450	The Name of the Rose	1983	£5.95	07/07/1999	
439-2	254	The Return of the Naked Chef	2000	£24.80	23/10/2020	
597-7	202	The BFG	1982	£3.99	05/01/2000	
624-t	234	Northanger Abbey	1855	£5.95	01/03/2003	
624-X	230	Harry Potter and the Goblet of Fire	2000	£5.95	16/03/2003	
625-X	334	Sense an Sensibility	1876	£6.95	01/03/2003	
629-0	340	Harry Potter and the Prisoner of Azkaban	1999	£7.50	19/04/2001	
670-8	160	Animal Farm	1945	£6.99	03/04/1999	
712-4	212	Galileo's Daughter	1999	£10.40	03/04/1999	
898-3	250	Chocolat	1999	£6.99	01/05/2000	

Use the ISBN as your primary key.

Once all the information has been entered, close the table and name it 'books' table.

Next, let's create a table for Authors. Using the 'Create Table in Design View'.

Field Name Column	Data Type
AuthID	AutoNumber. Auto number is an automatically assigned sequence number.
AuthLastName	Text
AuthFirstName	Text
Nationality	Text

Make the Author ID the primary key. Save the table as 'Authors'.

Open the table and add in the following information.

AuthID	AuthLastName	AuthFirstName	Nationality
1	Austen	Jane	British
2	Harris	Joanne	British
3	Marquez	Gabriel Garcia	Columbian
4	Rowling	J.K.	British
5	Alcott	Louisa May	American
6	Dahl	Ronald	British
7	Bronte	Charlotte	British
8	Dickens	Charles	British
9	Carey	Peter	Austrialian
10	Oliver	Jamie	British
11	Orwell	George	British
12	Proulx	E. Annie	American
13	Sobel	Dava	British
14	O'Connor	Joseph	Irish
15	Eco	Umberto	Italian

Close the table.

Delete a column from a table

Let us delete the comments column from the books database. There are 2 ways to delete this:

- 1 Datasheet view - right click on the column header and choose delete column, or you could;
- 2 Design view - click anywhere in the row you want to delete, choose Edit - Delete Row

Try deleting a column using the Datasheet view.

Adding a new field to table

You can add new fields to your database at anytime but it is however best to think about the fields you want to add when you are setting up the table and modify the design as little as possible after that.

Adding fields can be inserted in both the datasheet view as well as in design view. Let's see what happens when you have attempted steps 1 and 2.

- 1 Datasheet view - Lets insert the comments column back into the book table. Click anywhere in the table to the right of where you want to place the new column. Choose Insert - Column. Right click on the column header and choose Rename Column. Type the name Comments and then press enter.
- 2 Design View - Lets insert a new column - Recommended. This can be added to the end of your list. Type Recommended in the Field Name column and then select Yes/No in the Data Type column. If the column needs to be between two existing fields. The new row will appear above, so Click the row below. Choose Insert - Row. Type in Recommended and select Yes/No in the second column.

Once the Recommend column is added, return to your datasheet and tick about half the books from the table as recommended. If you have no personal preference, simply tick every second book. Save the table.

Now that you have practiced setting up a database, let's start to create table relationships.

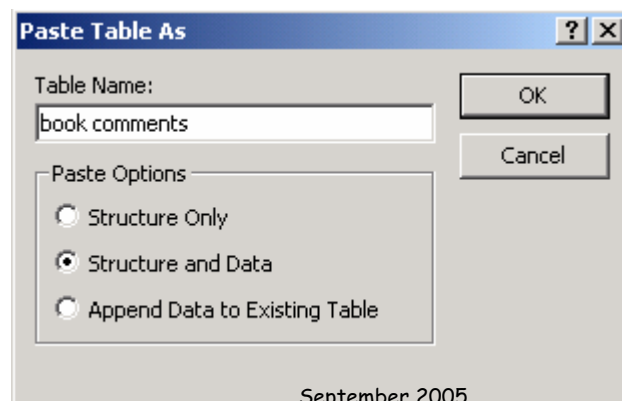
Create a one-to-one relationship

From your bookworm database, you decide that you want to separate the comments and recommendations from the rest of the table. Begin by creating a new table named 'Book Comments' to hold the Comments and Recommended fields. Then you will build a one-to-one relationship between the Books table and the new Book Comments table.

To do this, make a copy of the book table within the Bookworm database. Include both the structure and the data in the copy and name it Books Comments.

Copy a table within a database

Return to the main menu and right click on the books table, select copy. To copy the same database choose Edit - Paste. Access prompts you to name the copy, and you are offered a choice of what you want to copy. Type in book comments.



In most cases, the Structure and Data is the best option. This copies both the structure of the table and all the data.

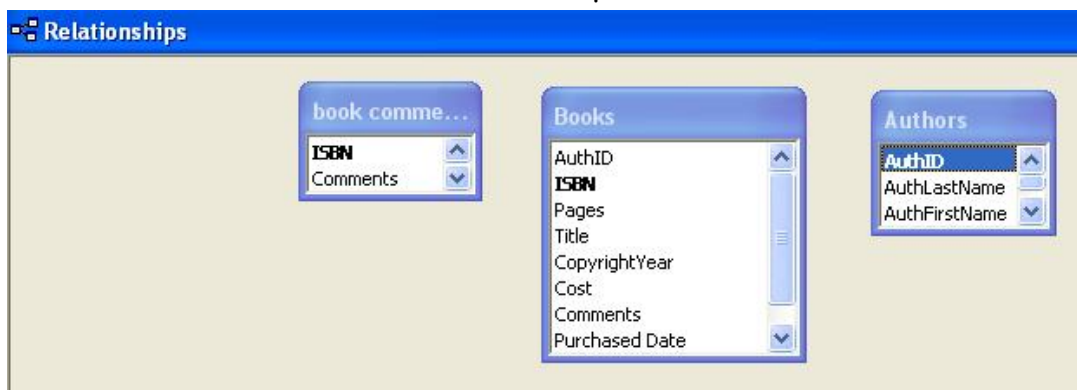
Structure Only - copies the structure of the table only, without any data.

Append Data to Existing Table - Access prompts you to enter the name of the table to which you want to append the data. The destination table must have the same structure as the table you are copying from.

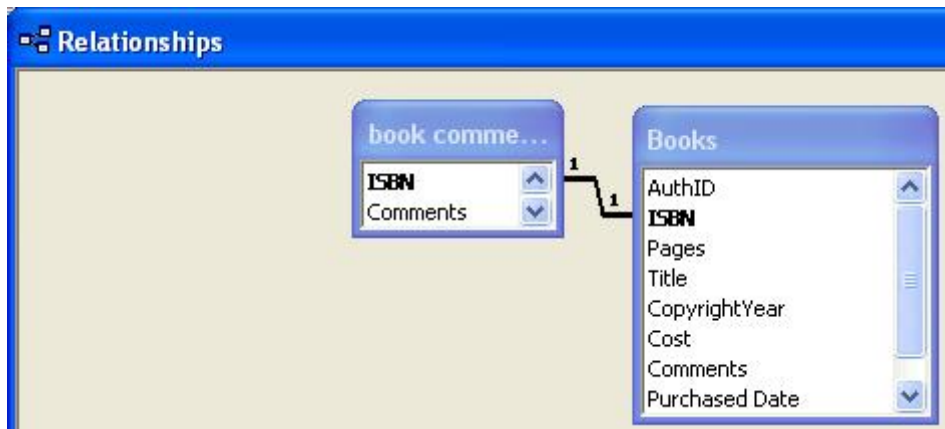
Your copied Book Comments table is now listed in the Tables tab of the database window. Double click it to open the datasheet view.

Delete the following columns from the Book Comments table - title, copyright Year, Purchase Date, Cost and Pages. Only the following fields should remain - ISBN, Comments and Recommended. Save and close the table

To create the one-to-one relationship from your Bookworm database window, choose Tools - Relationship to display the relationship window. You should see the three tables you have created in the Relationship window. If you don't see all three tables, click the Show Table button and add any missing table. You can change the position and size of the individual table windows as required.



In the Books table, click the ISBN field and hold down the mouse button. Notice how the mouse pointer changes shape. Drag the cursor over to the ISBN field in the Book Comments table and release the mouse button. In the Edit Relationship dialog box displayed, select the Enforced Referential Integrity option, and click the Create button. Access now displays a line joining the ISBN fields in the two tables - this represent the relationship you have just created.



Close the Relationship window and save the layout changes when prompted.

When you open the books table now, you can see a new column of plus (+) signs at the left of the table. When you click a + sign, Access displays the linked record from the Books Comments table. To close the window showing the linked record, click its minus (-) sign.

Books : Table											
	AuthID	ISBN	Pages	Title	Copyright Year						
-	1	010-9	302	Emma	1816						
	<table border="1"> <thead> <tr> <th>Comments</th> <th>Recommended</th> </tr> </thead> <tbody> <tr> <td></td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>*</td> <td><input type="checkbox"/></td> </tr> </tbody> </table>		Comments	Recommended		<input checked="" type="checkbox"/>	*	<input type="checkbox"/>			
Comments	Recommended										
	<input checked="" type="checkbox"/>										
*	<input type="checkbox"/>										
+	7	011-7	345	Jane Ayre	1847						
-	8	016-8	412	Great Expectations	1861						
	<table border="1"> <thead> <tr> <th>Comments</th> <th>Recommended</th> </tr> </thead> <tbody> <tr> <td></td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>*</td> <td><input type="checkbox"/></td> </tr> </tbody> </table>		Comments	Recommended		<input checked="" type="checkbox"/>	*	<input type="checkbox"/>			
Comments	Recommended										
	<input checked="" type="checkbox"/>										
*	<input type="checkbox"/>										
+	14	043-9	280	Cowboys and Indians	1991						
+	1	066-4	287	Mansfield Park	1814						
-	5	069-3	344	Little Women	1868						
	<table border="1"> <thead> <tr> <th>Comments</th> <th>Recommended</th> </tr> </thead> <tbody> <tr> <td></td> <td><input type="checkbox"/></td> </tr> <tr> <td>*</td> <td><input type="checkbox"/></td> </tr> </tbody> </table>		Comments	Recommended		<input type="checkbox"/>	*	<input type="checkbox"/>			
Comments	Recommended										
	<input type="checkbox"/>										
*	<input type="checkbox"/>										
+	12	242-4	289	The Shipping News	1993						
+	4	274-5	298	Harry Potter and the Philos	1997						

The Book Comments also contains plus (+) signs. When you click the + sign beside the ISBN, you will see the matching record from the books table.

book comments : Table						
	ISBN	Comments	Recommended			
+ 010-9			<input checked="" type="checkbox"/>			
- 011-7			<input type="checkbox"/>			
	AuthID	Pages	Title	Copyright Year	Cost	
	7	345	Jane Ayre	1847	£4.99	
*	0					
+ 016-8			<input checked="" type="checkbox"/>			
- 043-9			<input type="checkbox"/>			
	AuthID	Pages	Title	Copyright Year	Cost	
	14	280	Cowboys and Indians	1991	£8.99	
*	0					
+ 066-4			<input checked="" type="checkbox"/>			
+ 069-3			<input type="checkbox"/>			

Create a one-to-many relationship

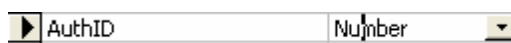
This is where a record in the first table i.e. authors can have many matching records in the second table i.e. books, but a record in the second table can have only one matching record in the first table.

Look at the Bookworm database you have created, some of the authors in the authors table wrote more than one of the books in the books table i.e. Rowling. But now books has more than one author.

One (Author)...	...to many (books)
Rowling	Harry Potter and the Philosopher's Stone
	Harry Potter and the Goblet of fire
	Harry Potter and the Prisoner of Azkaban

The primary key of the Authors table is Author ID. If an AuthID is added to the Books table, then books can be linked to their authors, and vice versa.

To do this, open the Books table in design view. Click in the second row and choose Insert - Row. Enter AuthID as the field name and select Number as the data type. Ensure that the Field Size property is set to Long Integer. Further now the panel set the Indexed field to NO. (This is explained later). Save the changes to the Authors table.



Open the datasheet view to enter the following Author ID details for each of the books.

Title	Author	AuthID
Emma	Austen	1
Jane Eyre	Bronte	7
Great Expectations	Dickens	8
Cowboys and Indians	O'Connor	14
Mansfield Park	Austen	1
Little Women	Alcot	5
The Shipping News	Proulx	12
Harry Potter and the Philosopher's Stone	Rowling	4
Love in the Time of Cholera	Marquez	3
The History of the Kelly Gang	Carey	9
The Name of the Rose	Eco	16
The Return of the Naked Chef	Oliver	10
The BFG	Dahl	6
Northanger Abbey	Bronte	7
Harry Potter and the Goblet of Fire	Rowling	4
Sense an Sensibility	Austen	1
Harry Potter and the Prisoner of Azkaban	Rowling	4
Animal Farm	Orwell	11
Galileo's Daughter	Sobel	13
Chocolat	Harris	2

Close the Books table.

To create the one-to-many-relationship between Authors and Books table. From the Bookworm database window, choose Tools - Relationships to display the Relationships window.

If you don't see your Authors and books table, click the Show Table button and add any missing table.

In the authors table, click the AuthID field and drag to the Author ID field in the Books table. Notice the cursor changes to a rectangle. Release the mouse button.

Access now displays the Edit Relationship dialog box. Select the Enforce Referential Integrity option. Click the Create button to close the dialog box and create the link. The one-to-many symbol (a sideways number 8) represents the on-to-many relationship between the tables. Close and save the table.

When you open the authors table and click the plus signs, they will link to the books table and display the list of books written by the author

Authors : Table						
	AuthID	AuthLastName	AuthFirstName	Nationality		
	+	1	Austen	Jane	British	
	+	2	Harris	Joanne	British	
	+	3	Marquez	Gabriel Garcia	Columbian	
	-	4	Rowling	J.K.	British	
		ISBN	Pages	Title	Copyright Year	Cost
	▶ +	274-5	298	Harry Potter and the Philos	1997	£4.9
	+	624-X	230	Harry Potter and the Goblet	2000	£5.9
	+	629-0	340	Harry Potter and the Prison	1999	£7.5
	*					
	+	5	Alcott	Louisa May	American	
	+	6	Dahl	Ronald	British	
	+	7	Bronte	Charlotte	British	
	+	8	Dickens	Charles	British	
	+	9	Carroll	Robert	Australian	

Primary Key and Foreign Key??

The author ID field is the primary key for the authors table. Within the Books table, however, it is a foreign key - this is the term used to describe a field in one table that refers to the primary key in another table.

The ISBN is the primary key in the books table, but it is a foreign key in the Books Comment table.

Delete a relationship

You can delete a relationship by choosing Tools - Relationships to display the tables. Right click at or near the centre point of the relationship line. Choose Delete and click Yes to confirm. Close the relationship window when finished.

Enforced Referential Integrity

It is a good idea to check the Enforce Referential Integrity box when linking tables. When a table is unlinked to another, you can work with it without reference to any data that is contained in any other table. But when a table is linked to another, you need to take the details of that relationship into account when working with either table. The restrictions that apply in such situation come under the title of referential integrity.

When referential integrity is enforced in tables, Access applies the following restrictions to relationships.

Books and Authors tables	In general
In the books table, you cannot enter an AuthID if that AuthID is not present in the Authors table.	You cannot enter a value in a foreign key field if that value doesn't exist in the primary key field of the related table.
In the Authors table, you cannot delete an author if that authors AuthID is referenced in the Books table.	You cannot delete a record from a table if foreign key values in a related table point to that record.
If the AuthID fields of the Authors table were not AutoNumbers but were editable by a user, an AuthID still could not be changed if that value is referenced in the Books table.	You cannot change a primary key value if there are linked valued in foreign key fields of a related table.

Cascade Options

A further two options appear when you check the Enforced Referential Integrity box. Cascade Update related fields and Cascade Delete Related Records. These allow you to perform actions restricted by referential integrity, while ensuring that access automatically takes action to prevent inconsistencies. As a general rule, these options are better left unselected.

Indexed field

Looking up a name in the index at the back of a book is easier than scanning pages of text to search for what you want. Indexes in a database table work in the same way. Table sorting and querying based on indexed fields take less time because Access needs to examine only the indexed fields rather than entire records.

Access automatically makes a table's primary key field an indexed field. You can verify that the ISBN field of your books table is indexed by displaying the table in design view, clicking the data type of the ISBN field and looking in the lower panel.

Indexed	Yes (No Duplicates)
---------	---------------------

Similarly, look in the Authors table and the AuthID field, this should appear in the same way.

Index non-primary key fields

You can index other fields other than the primary key, and you can decide whether or not you want to allow duplicate values in such fields. If you look back at your books table and clicked in the data type of AuthID you have entered NO.

No - (Default) no index

Yes (Duplicates OK) - the index allows duplicates

Yes (No Duplicates) - the index doesn't allow duplicates

Remove an index from a field

To remove an index from a field that is not the primary key, but not actually remove the field itself, select the field in Design view and set the Indexed field to NO.