Intermediate Microsoft Access 2007

Using More Than One Table

Often times a database will have more than one table. For example, in a Customer database, one table might contain information about each individual customer and another table might hold information about which items they ordered.

In order for the user to query information found in multiple tables, a relationship between the tables must be defined.

Understanding the Types of Relationships in an Access Database

One-To-Many Relationship
One record in a table is related to many records in another table. An example would be that one customer can place many orders.

A Many-to-Many Relationship
A good example of this would be between a Products table and an Orders table. A single order can include more than one product. On the other hand, a single product can appear on many orders.

A One-to-One Relationship
Each record in the first table can have only one matching record in the second table, and each record in the second table can have only one matching record in the first table.

Relationships Between Tables

To Create a Relationship Between Tables:
  1) Click the Database Tools menu item
  2) Select Relationships

To Define the Relationships between the Tables in your Database:
  1) Double click on the tables in the Show Tables window to add them to the Relationships window
  2) Close the Show Table window
  3) Left click and hold the mouse button down on the field that you want to relate to a field in another table.
  4) Drag your mouse from one field in one table to another field in another table.
Now displayed is the **Edit Relationships** dialog box.

![Edit Relationships dialog box](image)

**Parent/Child Table Relationships**

This is best understood by examining the relationships defined in the example displayed above.

In the **Courses Table** the Course ID field is the Primary Key field. For each record in the Courses Table, there are many related records in the **Enrollments Table**. In this example, the Courses Table is considered the **Parent Table** and the Enrollments Table is considered the **Child Table**.

In the **Employee Table**, the Primary Key field is Social Security Number. For each record in the Employees table, there are many related records in the Enrollment Table.

In this example, the Employees Table is considered the Parent Table and the Enrollments Table is considered the **Child Table**.

**Referential Integrity**

Enforcing referential integrity ensures that the following three rules will not be broken:

- You are unable to enter a value in the foreign key field of a child table if that value doesn’t exist in the primary key of the parent table.
- You are unable to delete a record from a parent table if matching records exist in a related table.
- You are not able to change the value in the primary key field if that record has related records in another table.
To Enforce Referential Integrity:

1) Click in the **Enforce Referential** Integrity check box
2) Click the **Create** button.

The relationships window will show a line between the related fields as shown below.

On one end of the line is a “1” indicating the “one” side of a one-to-many relationship.

The other end of the line will have an infinity symbol next to it to show that there are potentially many entries on the “many” side of the one-to-many relationship.

**Cascade Update Related Fields**
If this item is selected, the following rule applies:

Access will automatically update any foreign key values in the “child” table (the “many” table in a one-to-many relationship) if you change a primary key value in a “parent” table (the “one” table in a one-to-many relationship).

**Cascade Delete Related Records:**
If this item is selected, the following rule applies:

Record(s) in the child table will be deleted when a related record(s) in the parent row are deleted.
Creating Queries

Using AND Logic in a Query

Using **AND** logic in a Query will return records in which one condition is true AND another condition is also true. For example:

**To Find all Employees who Attended Access and Passed:**
1) Select the **Create** tab on the Ribbon
2) Click the **Query Design** Icon

In order to add fields to the Query Grid, you must first add the tables that the fields are in.

**To Add Tables to the Grid:**
1) Double click on the table names
2) Click the **Close** button in the Show Table window once you have added all the necessary tables

3) Double click to add a field to the Query Grid
4) In the Criteria Row, enter the criteria
The criteria entered in the example below are as follows:

Course Id field criteria: Access*
This will locate any Course ID that begins with the word Access and ends in anything.

Passed field criteria: Yes
This will locate any records of individuals who have passed Access

If both of these criteria exist on the same row of the Query, it creates **AND** logic.

To Run the Query:
Click the **Run** icon (Red Exclamation Point) on the Ribbon

The results of this Query are displayed below:
Saving a Query

When you save a Query, what you are saving is really the question you are asking, not the results that you see when you run a query. For example, in the above query we asked Access to display any records where the Course ID contains the word “Access” and the Passed field is “Yes”. If we run that query next week, we are likely to see more records as a result.

To Save the Query:
1) Click the Close Query icon
2) Click Yes to save the Query
3) Enter a name for the Query
4) Click OK

If necessary to view all Access objects in the Navigation Bar:
1) Click on the Navigation bar drop down arrow
2) Select All Access Objects

Creating OR Logic in a Query

OR logic is based on the fact that either condition(s) can be true in order for a record to be included in the results of the Query.
To Create a Query Using OR Logic:
1) Select the Create Tab on the Ribbon
2) Click the Query Design Icon
3) Double click on the table names to add the table(s) to the Query
4) Click the Close button in the Show Table window
5) Double click to add the field to the Query Grid
6) In the Criteria Row, enter the criteria for the Query. By placing criteria on two separate rows in the grid, OR logic is automatically created

Below is an example of OR logic in a query:
- Course Id field criteria: Access*
- Course ID Field criteria: Excel*
- Passed field criteria: Yes

This will locate any records of individuals who have attended Access or Excel and passed.

Below are the results of this query:
Using a Calculated Field in a Query

A Query can be used to perform calculations. As an example, you can create a new field in a Query that calculates the Hours Worked field multiplied by the Rate of Pay field. You are also easily able to add two fields together, or multiply one field by another field.

**To Create a Calculated Field:**
1) Click in the next empty field cell in the Query Grid
2) Press **Shift F2**
3) Type the name of the new field in the Query followed by a colon and then a space
4) Enter the formula: *Example HourlyRate: [cost]/[hours]*
5) Click **OK** when finished. This creates a new field in the Query called *HourlyRate.*

**To Format the Field to Currency:**
1) Right click on the calculated field you created
2) Select **Properties**
3) Click in the **Format** line
4) Select **Currency**
5) Close the Property Sheet
6) Click the Run Query Icon

Below are the results from the query:

![Query results](image)

### Creating a Summary Query

To summarize records in a table, use the **Group By** and **Sum** function in a Query.

### To Summarize a Field in a Query:

1) Click the **Totals** icon on the Ribbon
2) Notice the total row has now been inserted into the Query Grid
3) Select the **SUM** function for the field you would like to total

![Query grid](image)

4) Click the **Run Query** icon
Below are the results based on the criteria above:

![Database Query Interface]

**Including Operators in a Query**

You are able to create criteria based on the following operators:
- `< Less Than`
- `> Greater Than`
- `< Less Than or Equal To`
- `> Greater Than or Equal To`
- `= Equal To`
- `<> Not Equal To`

You are also able to define criteria based on a range.

**To Locate Specific Records Within a Range of Dates:**
1) Add the necessary fields to the Query Grid
2) After adding the fields to the Query Grid, position your cursor in the criteria row for the field you are defining.
3) Enter the criteria

In the example below, the criteria is to identify records where the registration date is between February 1, 2006 and February 28, 2006.

![Range Criteria Example]
Creating and Using a Parameter Query

In a Parameter Query the criterion written on the criteria line is actually a prompt to the user of the Query to type in a criterion. The prompt must be enclosed in square brackets (located above and to the left of the ENTER key) on the criteria line. When executed, the Query will display a prompt, and the user will enter the criterion he or she wishes to use. When the Parameter Query is run, the prompt written on the criteria line will display as shown below.

To Create a Parameter Query:
1) Position your cursor in the criteria row for the field that you would like to create the parameter for
2) Type [enter text to be displayed]
3) Press Enter when finished
4) Click the Run Query icon

The Query will return records in which the field contains the item the user entered.

Below is an example of a Parameter Query that prompts the user to enter a Course ID:
Below are the results for all records where the Course ID is Excel1:

Forms

Forms are used to either view records that are in a table, or used as a way to enter records into a table.

To Create a Form:
1) Select the table from the Navigation Pane that the form will be based on
2) Select the Create Tab
3) Choose Form

You are now in Layout View of the form.

There are 3 basic views in an Access form:

Layout View
The form is actually “live” in this view, so you can see your data as it will appear in Form view. However, you can also make changes to the form design in this view. Because you can see the data while you are modifying the form, this is a very useful view for setting the size of controls or performing almost any other task that affects the appearance and usability of the form.

Design View
This view gives you a more detailed view of the structure of your form. You can see the header, detail, and footer sections for the form. The form is not actually running when it is shown in Design view, so you cannot see the underlying data while you are making design changes; however, there are certain tasks you can perform more easily in Design view than in Layout view. In this view you can add controls to your form, edit control sources and resize form sections.

Form View
This view is used to enter, edit or display data in your table.
There are 3 main sections of an Access form:

These are the **Header, Detail and Footer**. If data is in the Form Header, the data will display/print at the top of the form only. Data in the Detail section is records from the table or query that the form is based on. If there is data in the Footer section, the data will display/print at the bottom of the form.

**Understanding Controls on a Form**

Forms are made up of 3 things: **Bound Controls, Unbound Controls** and **Calculated Controls**. A **Bound Control** is a control whose source is connected to a field or query. An **Unbound Control** is a control that is not connected to a source such as a field in a table or query. An example of this type of control is a title on a form, a picture, line or other label. A **Calculated Control** is a control whose source is a formula or expression.

**To Edit a Control on a Form:**
Switch to **Design View**.
You are able to edit, add and remove controls from a form.

On a form, there are two types of commonly added controls: a label control and text box control. A label control will allow you to simply add text to a form. A text box control will allow you to add a field from a table/query.

To Add a Label Control to a Form:
1) Switch to Design View if necessary
2) Click the Label Icon
3) Click your mouse where you would like to add the text
4) Type the text

To Add a Text Control to a Form:
1) Switch to Design View if necessary
2) Click on the Text Box icon on the toolbar
3) Click on the form where you would like the control to exist

Below is a picture of a text box control that has been added to a form:

To Edit the Label to the Left of the Control you just Added:
1) Double click on the Label Control
2) Type text

To Move a Control:
1) Select the Arrange tab on the Ribbon
2) Click on the control you wish to move
3) Click the Remove button from the Control Layout section of the Ribbon
4) Click and drag the control to a new location on the form
To Change the Tab Order on a Form:
1) Right click on the form
2) Choose **Tab Order**
3) Click to the left of the field
4) Click and drag the control to a new position in the Tab Order window

![Tab Order window](image)

To Add a Calculated Control to a Form:
1) Click on the **Text Box Control** icon
2) Click on the location on the form where you want to add the new control
3) Right click where it states unbound
4) Choose **Properties**
5) On the **Data** tab, click to position your cursor in the Control Source field
6) Enter a formula
7) Close the Property Sheet Window

![Property Sheet window](image)
Reports

In order to capture that data that you would like to report on, you must first create a Query to identify the records to be included in the report. After you have created the Query, you need to think about the design and layout of the report.

To Create a Report Based on a Query:
1) Click the Create tab on the Ribbon
2) Select Report Wizard
3) Select the table/query that you would like to base your report off
4) Select the field(s) and click the > sign to add the field to the report.
5) Click Next
6) Select a field to group by (optional)
7) Click **Next**
8) Select the field you would like to sort by
9) Select a layout and orientation
10) Click **Next**
11) Select a style for your report
12) Click **Next**
13) Enter a title for your report
14) Click **Finish**