Microsoft Access XP
Queries

Student Manual

the Abreon group

Foster Plaza 10
680 Andersen Drive
Suite 500
Pittsburgh, PA 15220
412.539.1800
800.338.5185

Table of Contents

Table of Contents

the Abreon group

Objectives

This course builds on the Access Introduction course to focus specifically on Access queries. Concepts from the Access Introduction course will be reviewed, as well as new topics will be introduced. These topics include:

- Understanding the use and role of queries in a database

- Designing and modifying selection queries

- Sorting records in a query

- Exporting query results to Excel and Word

- Setting criteria for a query

- Adding and deleting tables in a query

- Using AND, OR, and other conditional operators as criteria

- Using parameter queries to input criteria

- Using the Query Wizards to design queries

- Creating horizontal totals in queries

- Creating group total calculations in a query

- Using action queries to perform actions on tables.

Chapter 1

Creating and Using Queries

Chapter Objectives:

- Understanding Access Query Types

- Creating a Query

- Previewing and Running a Query

- Adding, Moving, and Deleting Fields

- Sorting Records in a Query

- Saving a Query

- Renaming a Query

- Exporting Query Results to Excel or Word

Overview

A query asks a question about information stored in a database table. A selection query is the most common type. It allows the user to view, analyze and even make changes to information in the tables and fields that were selected.

When a selection query is run, the answer is displayed in a dynaset that looks like a table in datasheet view. It is the output from the query. When the query is closed, the dynaset is not saved and no longer exists.

Since queries can be used over and over to review ever changing data, the query definition is named and saved. As the information in the table is edited or updated, the same question can be asked again, and the query will produce different results.

Why use a query?

- Only fields and records that meet the criteria will be included in the dynaset.

- The selected records can be sorted to appear in a specified order.

- The query can retrieve information from several tables. This alleviates the need to search through multiple tables manually.

- Reports and forms can be built upon queries so that every time the report is opened, the most recent information is displayed.

- When a query is saved the dynaset is not saved and no longer exists, only the commands which comprise the query are saved. Therefore, each time a query is run, it will provide the latest data from the underlying tables.

- In a query, the resulting dynaset is the actual data in the underlying tables. Changing the information in the dynaset in Datasheet view will change the data in the underlying tables.

- Fields can also be created which are mathematical computations of fields in underlying tables. This allows more efficient table design by allowing calculated fields to exist only as query statements rather than taking up space in every record of a table.

Chapter 1 – Creating and Using Queries

Understanding Access Query Types

Access provides the user with a number of different query types. Select queries only display the results of the query "question"; they make no modifications to the data. Action queries will modify the data in the underlying table(s) based on the criteria set in the query.

The following are the query types available in Access:

Selection Queries		
Select		Extracts data from one or more tables and returns values in a **dynaset**.
Parameter		Prompts the user for specific criteria each time the query is run. You can also create a custom form or dialog box that passes the criteria to the query.
Total		Provides the capability to sum or produce totals (such as sum or count) in a select query.

Crosstab Queries		
Crosstab		Produces a "spreadsheet" summary of data in which you specify the fields to be used as row and column headings.

Action Queries		
Make-Table Query		Creates a new table from all or part of the data in one or more tables.
Update Query		Makes global changes to a group of records in one or more tables.
Append Query		Adds a group of records from one or more tables to the end of another table.
Delete Query		Deletes a group of records from one or more tables.

the Abreon group

Creating a Query

Access uses a feature called Graphical Query by Example (QBE) to construct queries. QBE allows the user to create very complex database queries without the need for query language coding. Access QBE allows for very easy design of queries through the use of drag-and-drop, field selectors, and drop-down combo boxes.

To create a query, the user can drag and drop fields from the upper portion of the window into the grid located in the lower portion of the query window, or they can click the down arrow in the **Field** cell of the grid.

Each column of the grid contains information about a particular field included in the query. The **Query Design** window is where the selected query fields are then ordered, sorted, and criteria is specified. This is also where expression fields are created.

The following procedure illustrates the basic steps in creating a selection query. Further procedures in this course will explain the creation of criteria, expressions, and calculations.
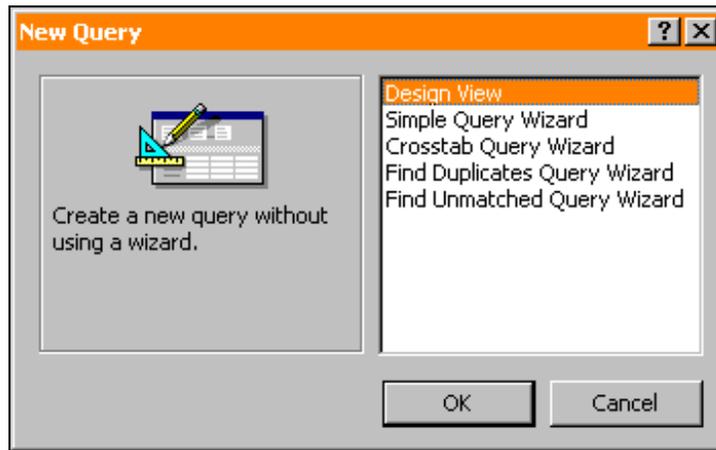
Procedure:

1. In the **Database** window, select the **Queries** tab.
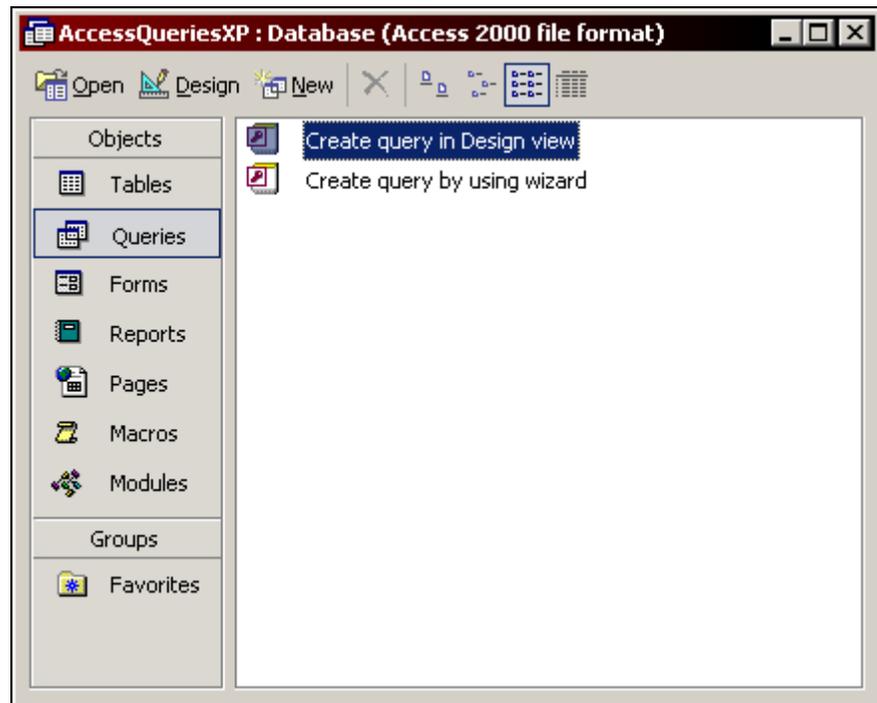
2. Click **New**. The **New Query** dialog box displays:

3. Choose **Design View** and click **OK**.

4. Access opens a new query window and displays the **Show Table** dialog box. This dialog box will list all the tables and queries currently in the database. Using the tabs at the top of the dialog box, you have the option of viewing only tables, only queries, or both.
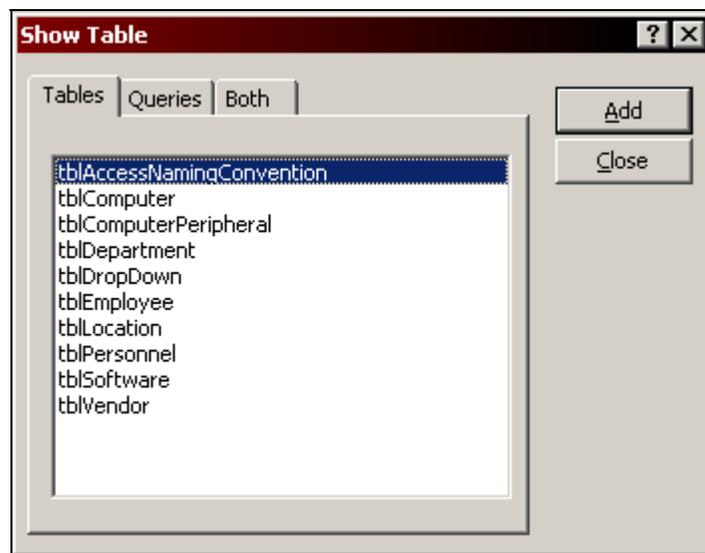
5. Select the table or query to be included in the new query and click **Add**. To include more than one table, repeat this step until all tables are selected. Click **Close** when finished. The **Query Design** window displays the selected objects in the upper portion of the window.
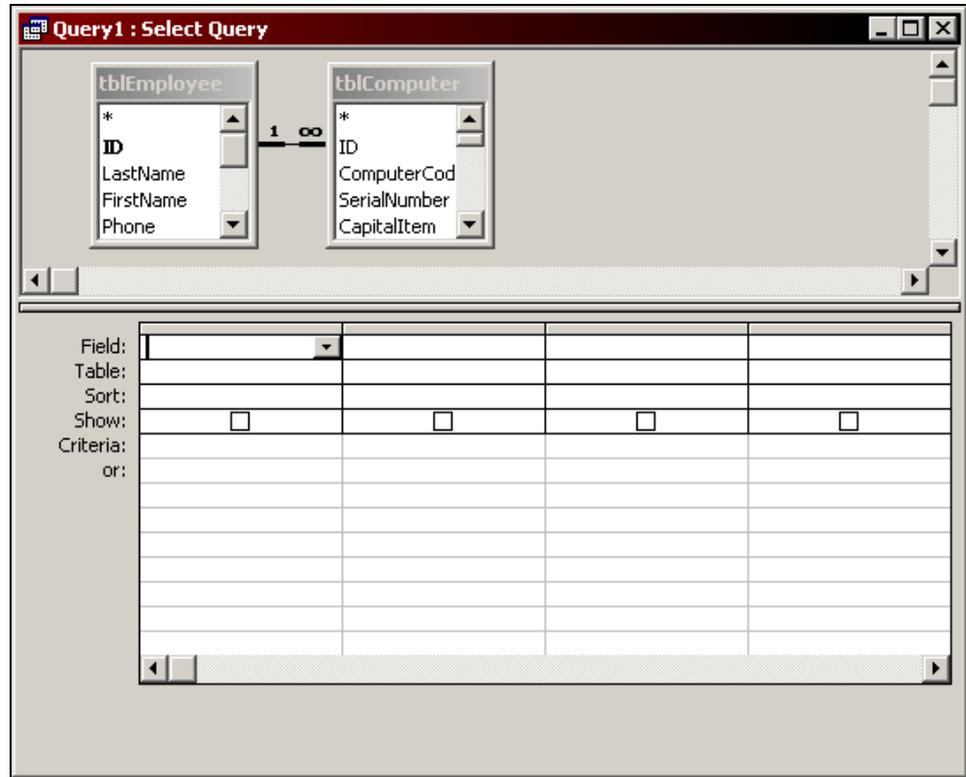
6. Specify which field (or fields) will appear in the dynaset by dragging the field name from the field list window down into the grid.

    OR

    Use the **Table** and **Field** drop-down lists in the lower portion of the window to select fields.

    OR

    Double-click the field name in the field list window.

7. To select a continuous range of fields, click the first field, press and hold the **[Shift]** key, click the last field, and then drag the highlighted list.

8. To select multiple random fields in a single table, press and hold the **[Ctrl]** key, click the desired fields, and drag the entire highlighted list onto the lower portion of the window.

9. To insert all fields from a field list window into the grid, *double*-click the title bar of the field list window and then drag any field to the grid.
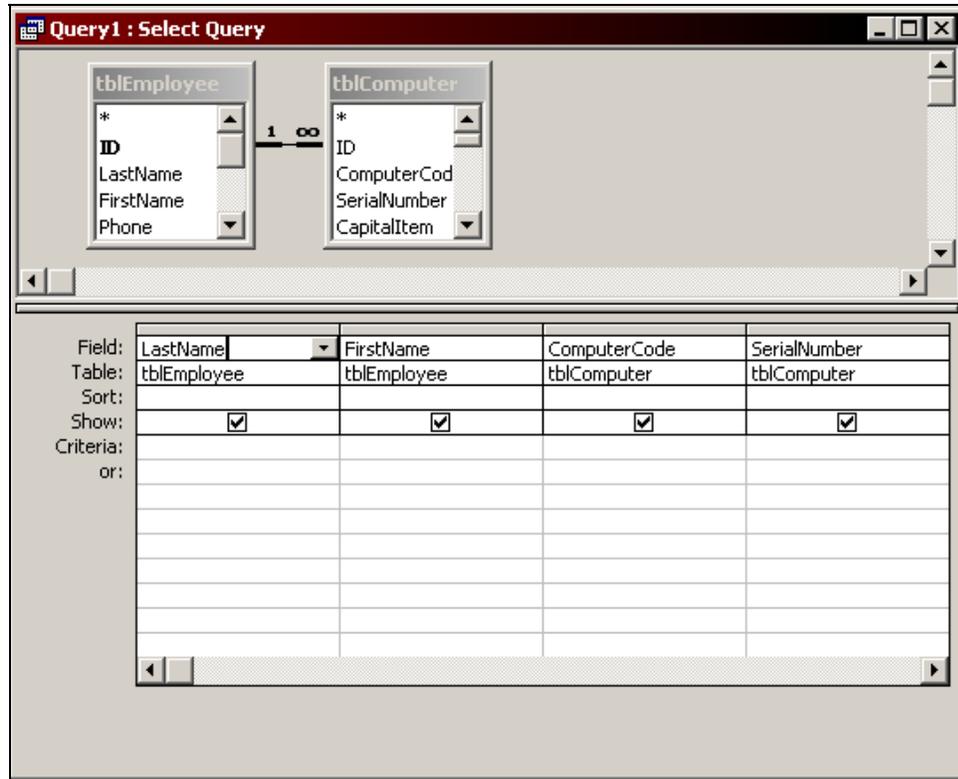
10. To sort the records, select the **Sort** cell below the field's name. Click the drop-down arrow that appears to view the sorting choices: **Ascending** (0-9 and A-Z), **Descending** (9-0 and Z-A), or **(not sorted)**.

11. Add any expression fields based upon selected tables. Expressions must follow certain rules:

    • Field names must appear in brackets [ ].

    • The expression is not case sensitive.

    • The available operators are " + " (addition), " - " (subtraction), " * " (multiplication), and " / " (division).

12. If desired, clear the **Show** check box under a field name to have that field NOT appear in the resulting dynaset.

13. Set up any criteria required to achieve the desired dynaset. Use the following guidelines for creating criteria:

the Abreon group

- Text must appear in double quotes (")

- Dates will appear between number signs (#)

- Wild cards (***** and **?**) may be used

- Criteria can be entered which require the records retrieved to meet *all* the criteria (AND operators).

- Criteria can be entered which require the records retrieved to meet only *one* of the specified criteria (OR operators).
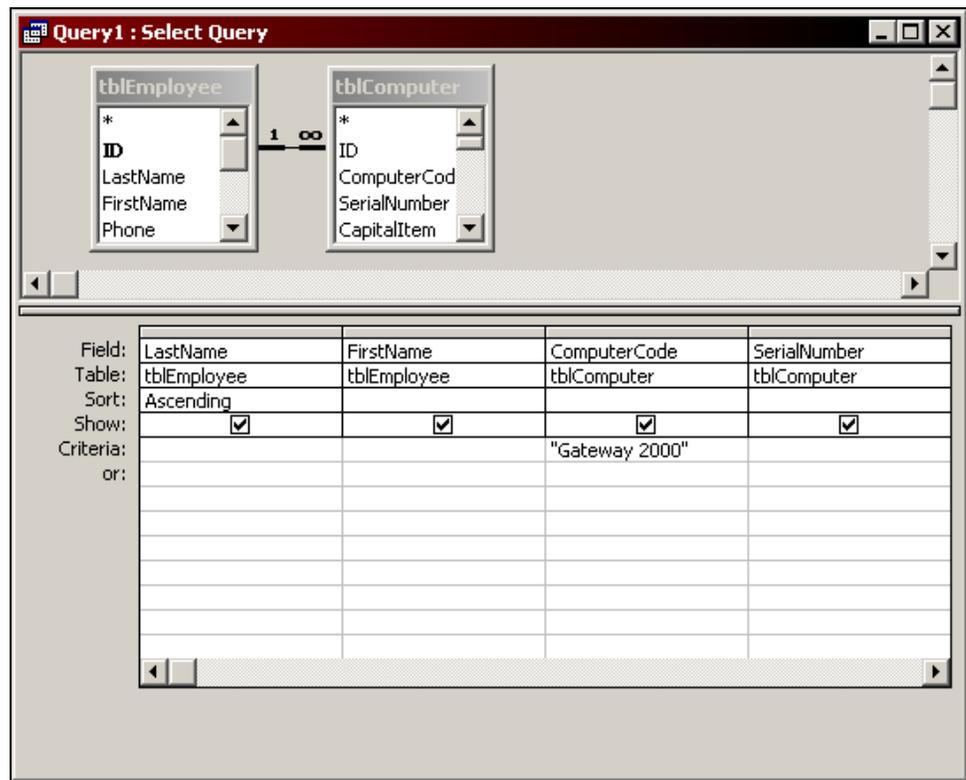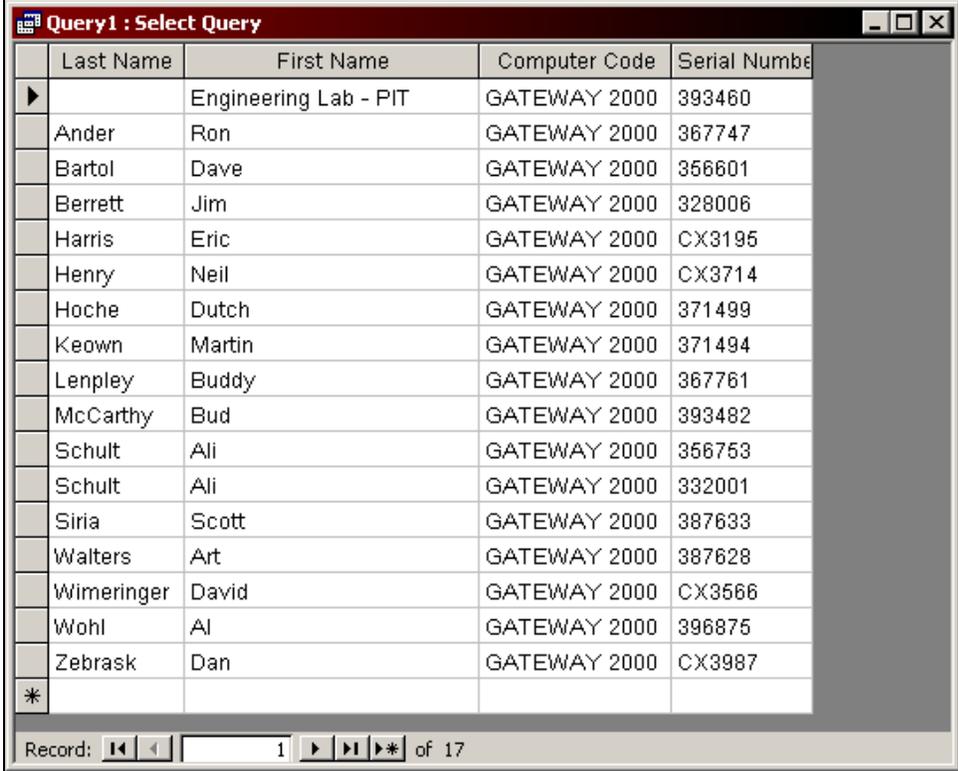

14. Select the **Query** menu and choose **Run**.

OR

Click the **Run** button on the **Query Design** toolbar to execute the query and view the results.

Last Name	First Name	Computer Code	Serial Numbe
	Engineering Lab - PIT	GATEWAY 2000	393460
Ander	Ron	GATEWAY 2000	367747
Bartol	Dave	GATEWAY 2000	356601
Berrett	Jim	GATEWAY 2000	328006
Harris	Eric	GATEWAY 2000	CX3195
Henry	Neil	GATEWAY 2000	CX3714
Hoche	Dutch	GATEWAY 2000	371499
Keown	Martin	GATEWAY 2000	371494
Lenpley	Buddy	GATEWAY 2000	367761
McCarthy	Bud	GATEWAY 2000	393482
Schult	Ali	GATEWAY 2000	356753
Schult	Ali	GATEWAY 2000	332001
Siria	Scott	GATEWAY 2000	387633
Walters	Art	GATEWAY 2000	387628
Wimeringer	David	GATEWAY 2000	CX3566
Wohl	Al	GATEWAY 2000	396875
Zebrask	Dan	GATEWAY 2000	CX3987

Record: 1 of 17

Tips:

- If more than one sort is selected in the QBE grid, the fields are sorted in the order they appear in the query grid (left to right).

- The **Show** option is useful for sorting by a specific field or specifying criteria that you do not wish to have displayed in the resulting dynaset.

Previewing and Running a Query

The results of a query can be obtained in one of two ways. Either a preview of the results can be viewed in **Datasheet view**, or the query can actually be **Run** or executed.

In the case of selection queries and crosstab queries, there is no difference between viewing a query in Datasheet view or actually running the query.

However, with the following action queries types – append, delete, and make-table – previewing a query allows you to check that the query criteria and results are correct before performing the query action on the table data. It is not possible to preview the effects of an update query.

Procedure: (To preview a query in Datasheet view)

1. On the **Query Design** toolbar, click the **View** drop-down button and choose **Datasheet View**.

OR

Select the **View** menu and choose **Datasheet View**.

2. The query results will be displayed. In the case of an action query, this is a preview of the results.

Procedure: (To run a query)

1. On the **Query Design** toolbar, click the **Run** button to execute the query and view the results.

OR

Select the **Query** menu and choose **Run**.

2. The query results will be displayed. In the case of an action query, this command will perform the selected action on the table data.

Adding, Moving, and Deleting Fields

Fields will be displayed from left to right in the dynaset in the order they appear in the QBE grid. However, it may be necessary to modify the dynaset after viewing the results.

Use the following procedures to add, move, and delete fields from the query results.

Procedure: (To add fields)

1. Display the query in **Design** view.

2. To add a single field to the end of the grid, *double*-click the field name in the field list window.

3. To add a single field between other fields already in the grid, drag the field name to the grid and drop it. Notice that fields shift to the right to accommodate the insertion.

4. Repeat either step to add additional fields.

5. **Run** the query to view the new results.
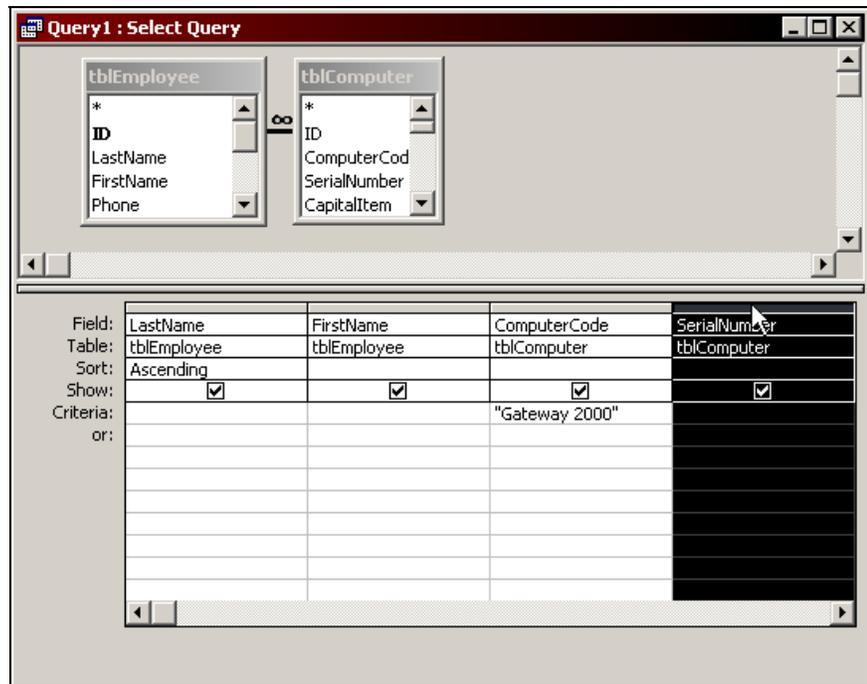
Procedure: (To move a field)

1. Display the query in **Design** view.

2. Position the pointer on the field's column selector at the top of the column in the QBE grid. The pointer changes to a small downward arrow.
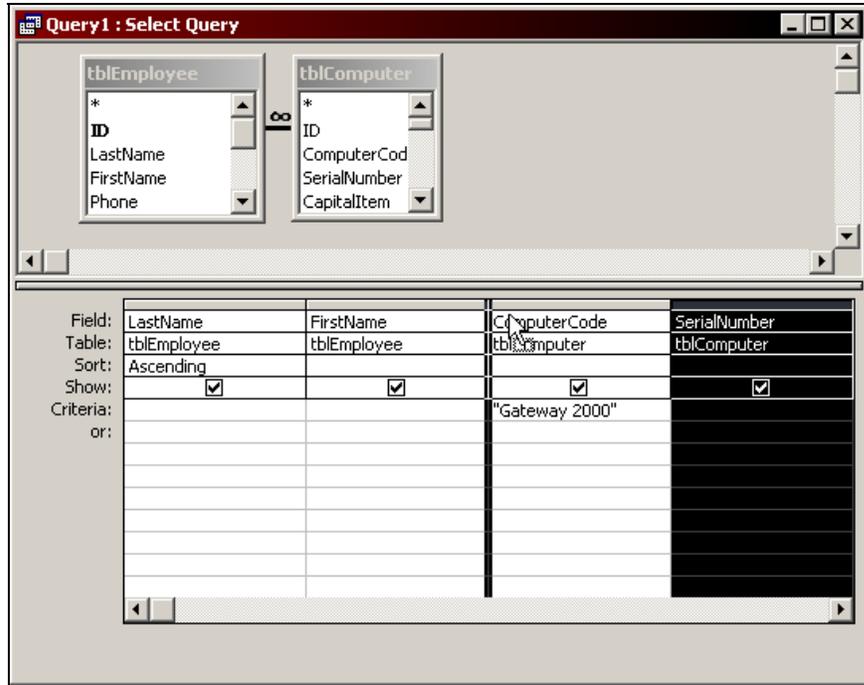
the Abreon group

Field column
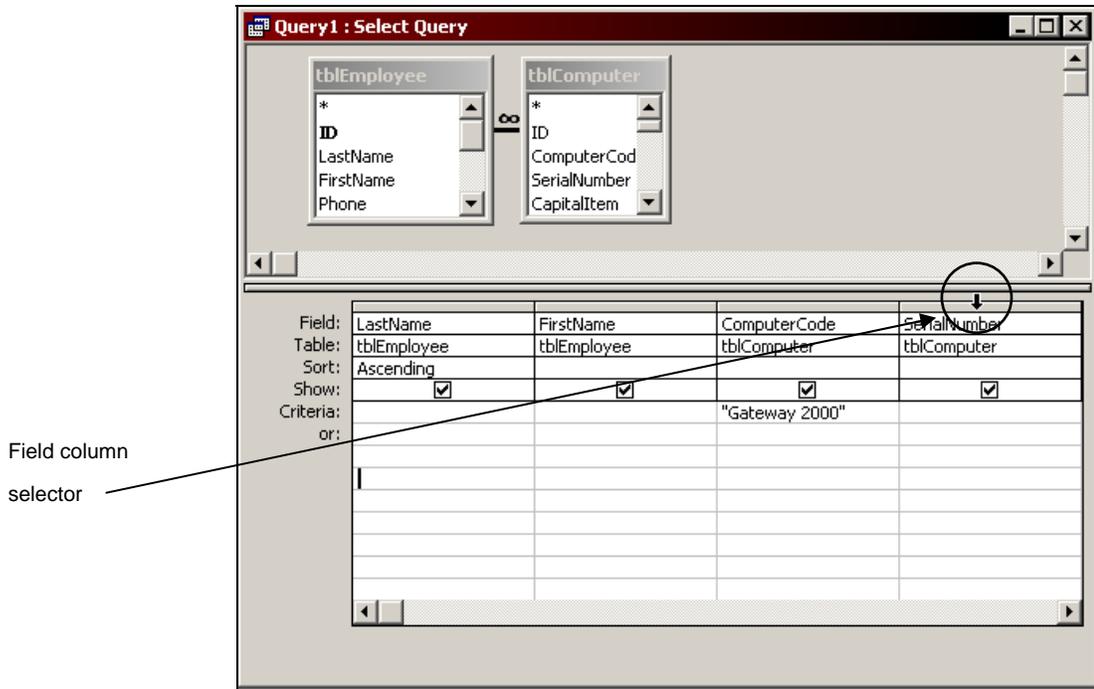
selector

3.   Click once to select the column. The pointer shape changes to a leftward pointing arrow.
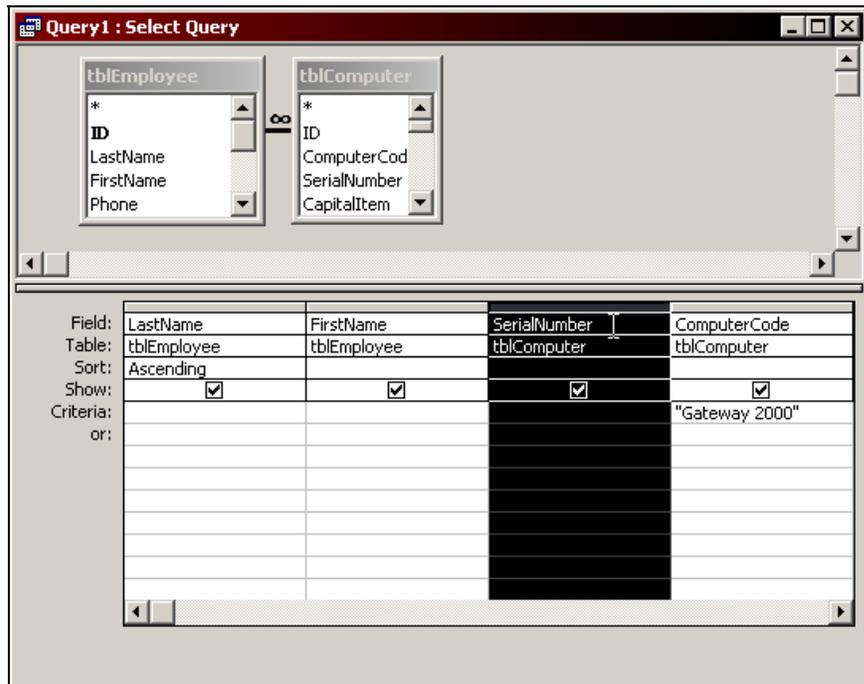

4.   KEEPING THE MOUSE IN THE SAME SPOT, press and drag the column selector. As you drag along the grid, a black line will appear between the columns to indicate where the field will be inserted.
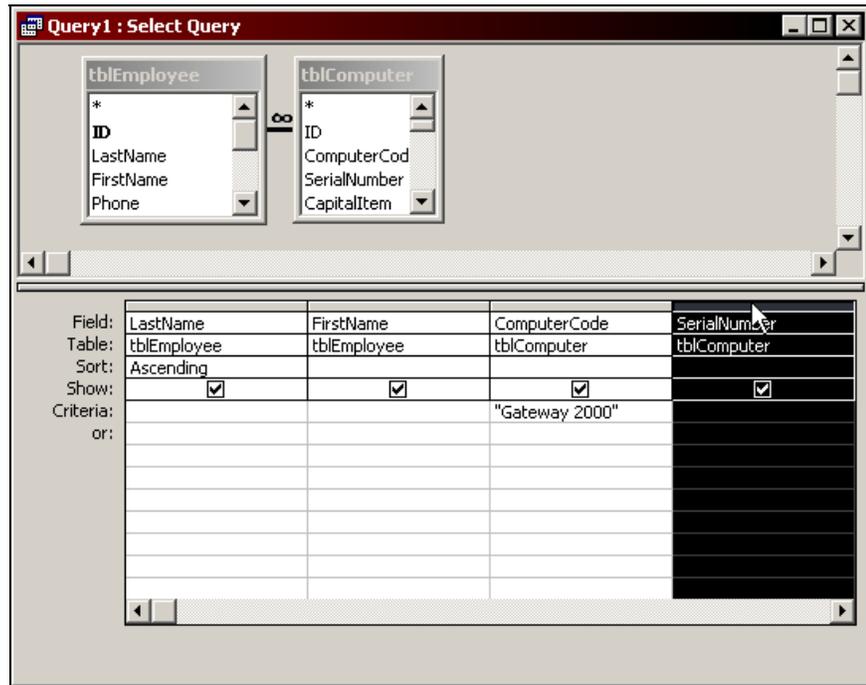
5.   Release the mouse button when the field is in its new location.

Procedure: (To delete a field)

1. Position the pointer along the field's column selector. When the pointer becomes a downward arrow, click to select the entire column.

2. Press the **[Delete]** key.

Tip:

*   You can clear the entire QBE grid by using the **Clear Grid** command on the **Edit** menu.

Sorting Records in a Query

Access displays records in the dynaset in the same order in which they appear in the underlying table or query. For easier editing and review, the records can also be sorted alphabetically, numerically, or chronologically. This can be done either in the query design view, or in the datasheet view.

Procedure: (To sort records in Query Design view)

1. Display the query in **Design** view.

2. Click in the **Sort** cell below the field's name. A drop-down arrow and menu will appear.
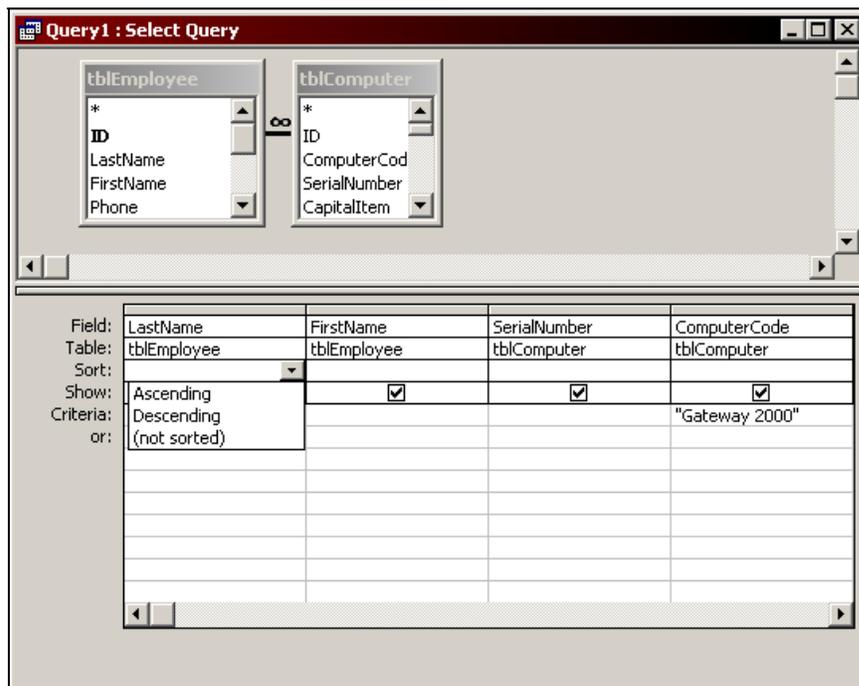
3. Select a sorting choice from the list: **Ascending** (0 - 9 and A -Z), **Descending** (9 - 0 and Z - A), or **(not sorted)**. If the **Sort** option is added to more than one field, Access will perform the sorts in order from left to right in the QBE grid.

4. Click the **Run** button on the **Query Design** toolbar to execute the query and view the results.

Procedure: (To sort records in Datasheet view)

1. Display the query in **Datasheet** view.

2. Click once in any cell in the field you wish to sort.

3. ⏏ Click the **Sort Ascending** or ⏏ **Sort Descending** button on the **Query Datasheet** toolbar.

Procedure: (To remove a sort in Datasheet view)

1. Display the query in **Datasheet** view.

2. Select the **Records** menu and choose **Remove Filter/Sort**.

Tip:

- It may be necessary to change the left-to-right order of the fields in the QBE grid to accommodate sorting. Once the query is run, the fields can be rearranged in Datasheet view to accommodate the desired layout of the results.

Saving a Query

To save a new query at any time, use the **Save** command. Access will also automatically prompt to save if a query window is closed without saving, or if changes are made to an existing query.

The **Save As** command can be used to create a copy of a query within the same database. This can be handy for creating similar queries without having to start over each time.

See **Appendix A – LNC Database Object Tags** for information on the recommended naming scheme for queries in a database.

Procedure: (To save a new query)

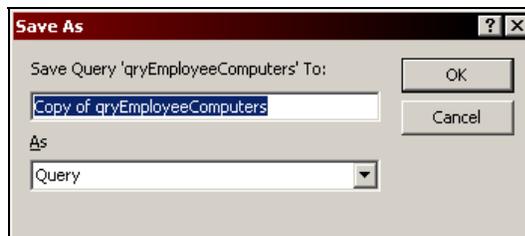1. Select the **File** menu and choose **Save**. The **Save As** dialog box displays:

2. Type a name for the query and click **OK**.

Procedure: (To create a copy of a query)

1. Select the query in the database window.

2. Select the **File** menu and choose **Save As**. The **Save As** dialog box displays:

3. Type a new name for the query and click **OK**.

Renaming a Query

It may be necessary to rename queries once they have been created. Renaming a query (or any Access object) is essentially the same as renaming a file in Windows – select the query name in the database window and then click again to edit the name, or *right*-click on the query name and select **Rename**.

However, caution should be exercised when renaming queries that serve as the basis for other queries, forms, or reports. Renaming the query may cause those objects to not function properly.

To prevent this from happening, Access provides the option to track name changes and make automatic updates using the **Name AutoCorrect** feature.

Procedure:

1. Select the **Tools** menu and choose **Options**.

2. In the **Options** dialog box, select the **General** tab.

3. In the **Name AutoCorrect** section, select the **Track Name AutoCorrect info** check box.
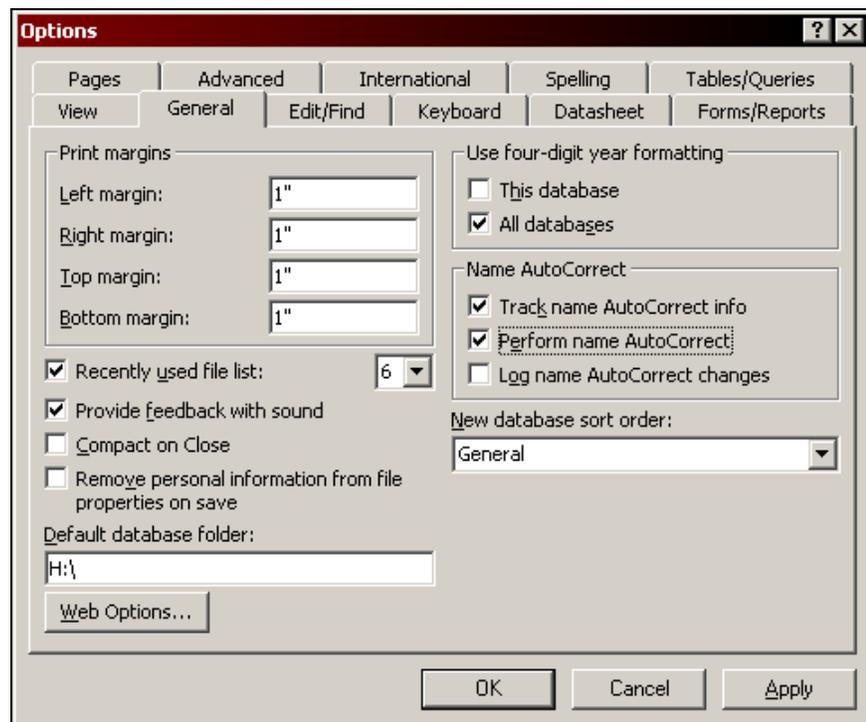
4.   To have Access automatically update all related objects when an object name is changed, select the **Perform name AutoCorrect** check box.

5.   To have Access maintain a log of all name changes, select the **Log name AutoCorrect changes** check box.

6.   Click **OK** to close the **Options** dialog box.

Tip:

- Although this feature appears in the **Tools, Options** menu, it is not a global feature for Access. It is a *per database* option, and must be activated for every database on which you wish to use it.

Exporting Query Results to Excel or Word

The results of a query can easily be exported to Microsoft Excel or Word. This ability allows for further analysis, formatting, or the use of the data in other documents.

Procedure:

1. **Design** a new query, or **Open**/**Run** an existing query.

2. When the results of the query are displayed, select the **Tools** menu, choose **Office Links** and then select one of the following options:
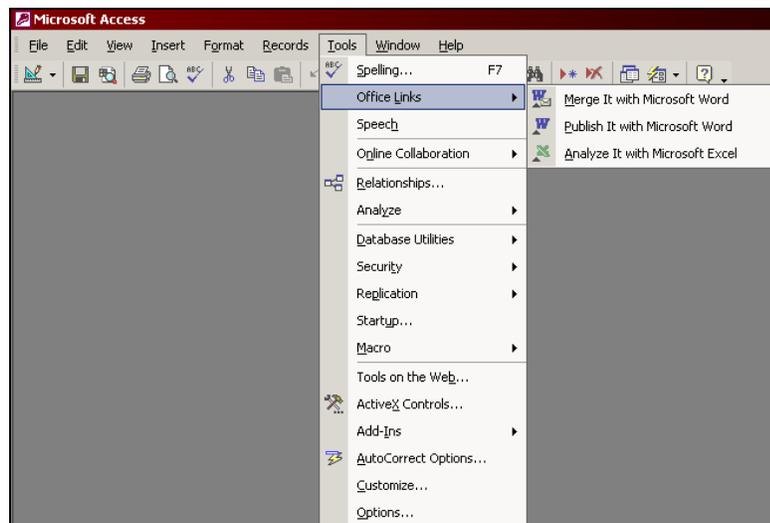
*   **Merge It with Microsoft Word** – this option allows you to use the query data as the data source for a Word merge document.

*   **Publish It with Microsoft Word** – this option exports the query results to a Word table.

*   **Analyze It with Microsoft Excel** – this option exports the query results to an Excel spreadsheet.

3. Access will output the results of the query as selected.

Tips:

- Changes made to data while in Excel or Word will not change the underlying data in Access.

- The Excel worksheet/workbook created by Access is initially created in a temporary directory. You should be save the query output to your usual document directory with a name other than that of the query. Use the **File, Save As** command to change directories and supply a new filename for the file.

- Some data types may be lost in the output to Excel. For instance, some number and time/date field types may not be recognized by Excel as their respective type, and will be treated by Excel as regular text. It may be necessary to apply cell formatting to correct this problem.

- Access will export the query data to Word in the Rich Text Format (.rtf). When saving the file in Word, it is advisable to change the file format to a standard Word document (.doc) using the **Save as Type** drop-down list in the **File, Save As** dialog box.

Chapter 2

Working with Queries

Chapter Objectives:

- Adding and Deleting Tables

- Setting Criteria for a Query

- Using AND and OR operators

- Using Conditional Operators as Criteria

- Using Joins in Queries

- Using Parameter Queries to Set Criteria

- Working with the Expression Builder

- Modifying Query Properties

Overview

In the first chapter of this book, the basic procedures of creating and working with a select query were explored.

In this chapter, you will learn how to take select queries further and work with more advanced query features, including establishing criteria, creating joins, using conditional operators, and changing query properties.

Adding and Deleting Tables

Although you can add more than one table when the query is initially created, you may want to add or delete tables later.
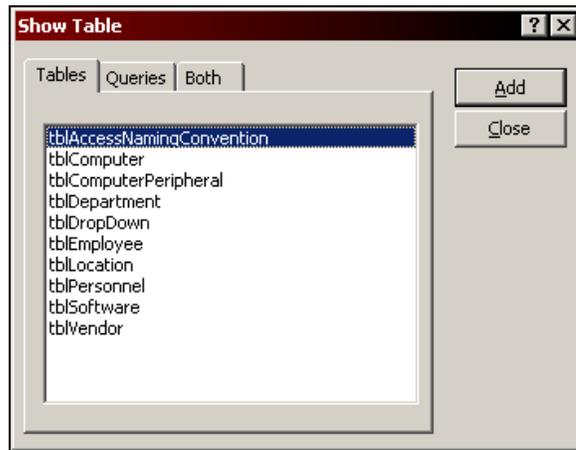
Procedure: (To add a table)

1. Click the **Show Table** button on the **Query Design** toolbar.

<div align="center">

OR

</div>

Select the **Query** menu and choose **Show Table**. The **Show Table** dialog box displays:

2. Select a table and click **Add**.

3. Repeat this process until all the tables are added. When completed, click **Close**.
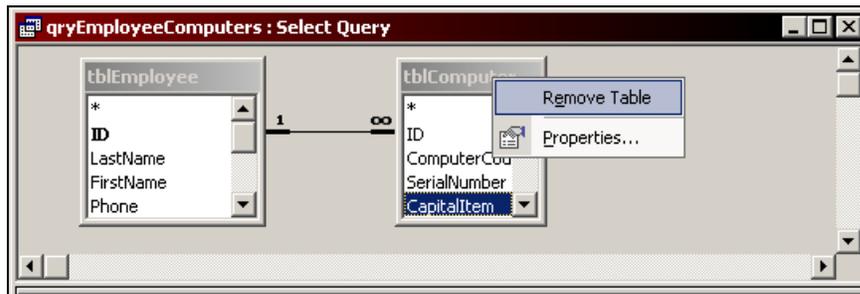
Procedure: (To delete tables from the query)

1. Select the unwanted table by clicking on its title bar.

2. Press the **[Delete]** key.

OR

Right-click the title bar and select **Remove Table**.

OR

Select the **Query** menu and choose **Remove Table**.

3. The table is deleted and all fields relating to that table are removed from the query grid.

Warning:

- Once deleted, the **Undo** command cannot be used to restore a table or field to a query. The table and fields will have to be added back in manually, or close the query without saving changes.

Setting Criteria for a Query

The QBE grid controls which fields are included in a query and in what order the records will appear. It is also possible to limit which records are included in the dynaset. This is accomplished by setting exact or general criteria for the query.

Exact criteria are used when the actual value is known, or you wish to limit your criteria to a single value. General criteria involve the use of a wildcard to select a range or group of values.

Procedure: (To set exact criteria)

1. Display the query in **Design** view.

2. Click in the **Criteria** cell below the field's name.

3. Type the value to be matched exactly. Text criteria will appear in double quotes ("), and dates will appear between number signs (#).

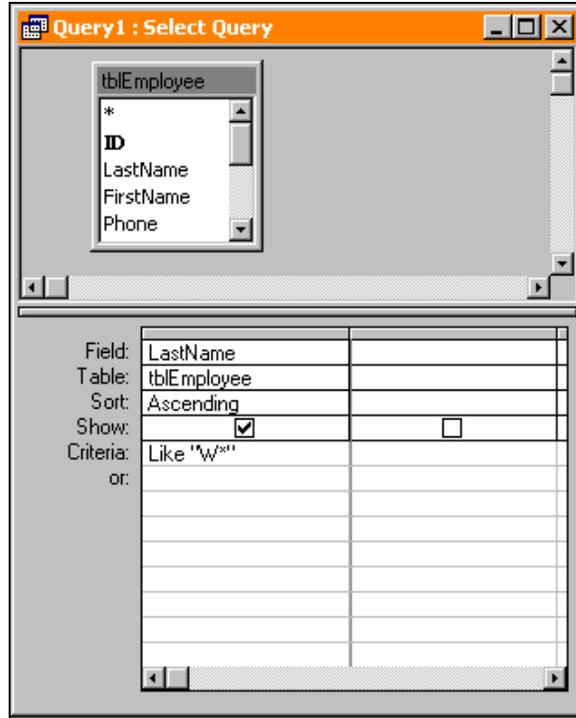4. **Run** the query to view the results.

Procedure: (To set general criteria)

1. Display the query in **Design** view.

2. Click in the **Criteria** cell below the field's name.

3. Enter the expression you want using a valid wildcard. The following table explains the wildcards and their usage:

Character	Description	Example Use and Results
*	Matches any number of characters. Can be the first or last character.	**wh*** finds what, white, and why.
?	Matches any single alphabetic character.	**b?ll** finds ball, bell, bill, and bull
[ ]	Matches any single character within the brackets.	**b[ae]ll** finds ball and bell, but not bill or bull.
!	Matches any character not in the brackets.	**b[!ae]ll** finds bill and bull, but not ball or bell.
-	Matches any one of a range of characters; must be in ascending order.	**b[a-c]d** finds bad, bbd, and bcd.
#	Matches any single numeric character	**1#3** finds 103, 113, 123

For example, to include only those employees with last names beginning with a W, type **w*** in the criteria cell of the **LastName** field.

4. **Run** the query to view the results.

Tip:

- When using wildcards in a query, Access will automatically add the word "Like" to the criteria.

Chapter 2 – Working with Queries

Using the AND and OR Operators

Queries retrieve records from the database if they meet the search criteria entered. You can enter a series of criteria and require that the records retrieved meet *all* of the criteria (AND criteria), or you can also enter a series of criteria and require that the records meet only *one* of the criteria (OR criteria).

For example, you can tell Access to retrieve employees who are salespeople AND living in Pittsburgh. The result is called the **intersection** of both sets of people. In an AND statement, the record must match **all** of the established criteria to be included in the resulting dynaset. Because of this fact, AND statements cannot be created in a single field.

For instance, you cannot create a query that lists employees who live in Pittsburgh AND Philadelphia. The field **City** can only have one value, and therefore, no single record can meet both conditions to be included in the query results.

You **can** however instruct Access to retrieve employees who live in Pittsburgh OR Philadelphia. The result is called the **union** of both sets of people. In an OR statement, the record only needs to meet any **one** of the established criteria to be included in the resulting dynaset. OR statements can be on the same field, or they can be between different fields.

In general, AND statements are created on the horizontal lines of the QBE grid, and OR statements are created in the vertical columns.

The two criteria on this horizontal line create an AND statement – "Pittsburgh AND Human Resources"

The two criteria in this column create an OR statement – "Pittsburgh OR Philadelphia"

the Abreon group

Procedure: (To create a union – "OR")

1. Display the query in **Design** view.

2. Click in the first **Criteria** cell of the field column for which criteria will be defined.

3. Type in the expression or value.

4. Repeat this step in the same column (using the additional **or** lines) for all other values to be included in the union.
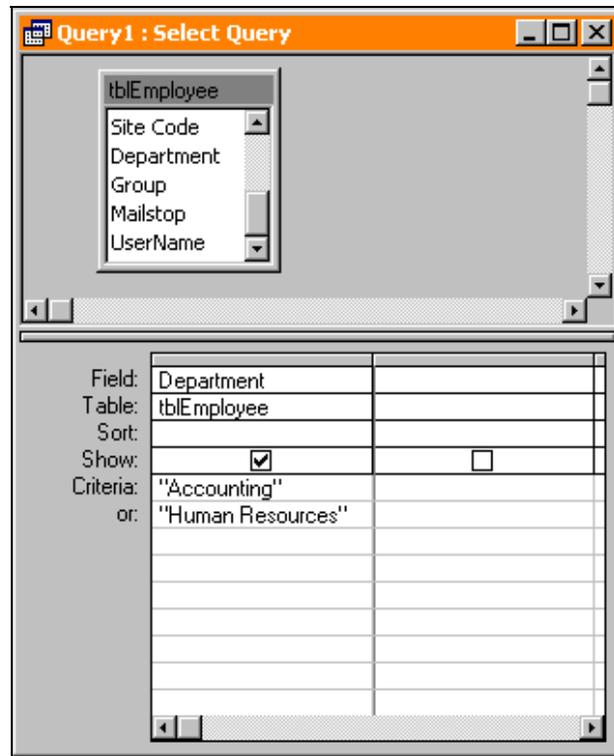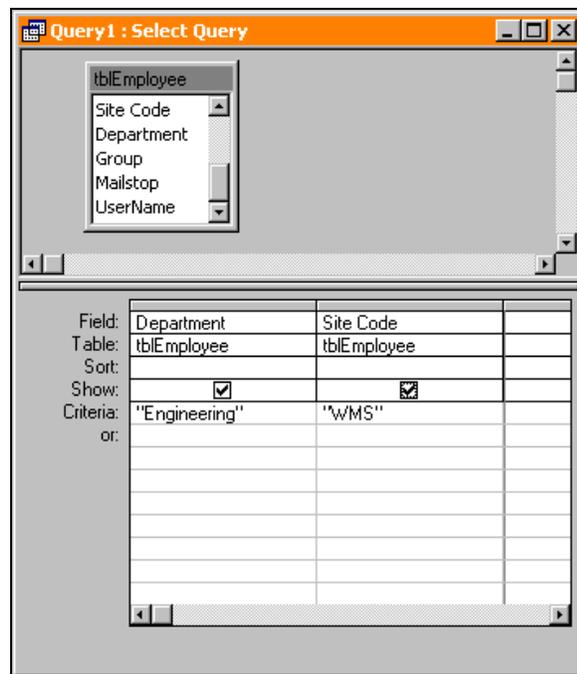

5. ![Run button] Click the **Run** button on the **Query Design** toolbar to execute the query and view the results.

Procedure: (To create an intersection – "AND")

1. Display the query in **Design** view.

2. Click in the first **Criteria** cell of the field column for which criteria will be defined.

3. Type in the expression or value.

4. Click in the first **Criteria** cell (on the same line) of the next field you want to specify criteria for.

5. Type in the expression or value.

6. Repeat the latter two steps for all other fields to be included in the intersection.

7. ![Run] Click the **Run** button on the **Query Design** toolbar to execute the query and view the results.

Tips:

- Multiple OR statements may be combined in a single cell. This is useful when you have two OR criteria for one field, and a single AND criteria for a second field, i.e. Department is "Accounting" OR "Human Resources" AND City is "Philadelphia".
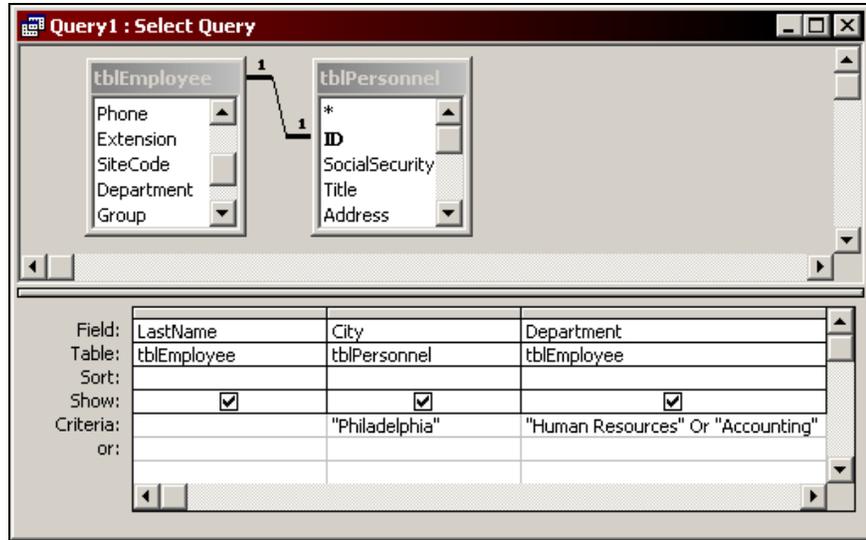
- OR statements can also be between different fields, however, the must appear on separate lines in the QBE or they will be interpreted as AND statements. In this case, the statement is "City is Philadelphia OR Department is Information Services".
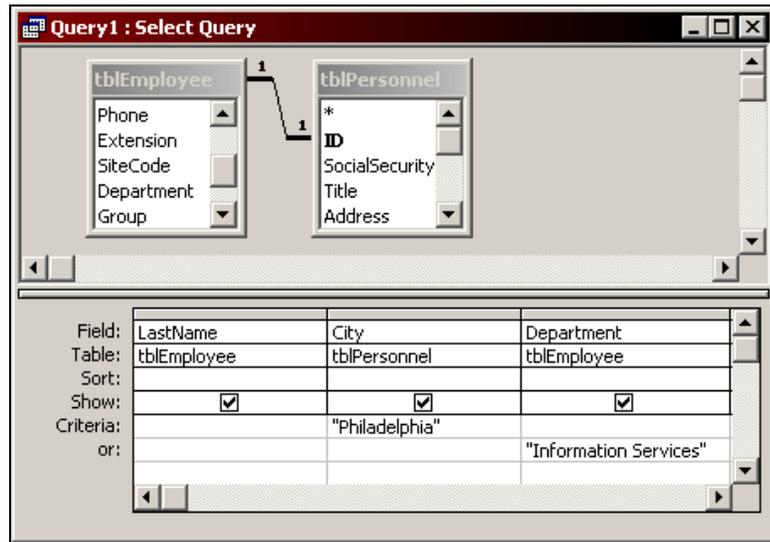
Using Conditional Operators as Criteria

The criteria section of the QBE grid allows many different types of conditions to be built into a query for sophisticated data analysis. The conditional operators are used in an expression to build those criteria.

Operator	Description	Criteria Example
*	Wildcard	T* (anything that starts with T)
?	Single character wildcard	T? (any 2 character item that starts with T)
#	Dates	#01/01/2000#
<	Less than...	< 100.00
>	Greater than...	> 100.00
<=	Less than or equal to...	<= 100.00
>=	Greater than or equal to...	>= 100.00
=	Equal to	= 100.00
<>	Not equal to	<> 100.00
" "	Literal string	"I want to match this string"
And	Logical and (must match both conditions)	[OrderAmount] < 100.00 And [Units] < 4
Or	Logical or (matching either condition = true)	[OrderAmount] < 100.00 Or [Units] < 4
Not	Logical not	Not [OrderAmount]
Between…And	Between two values (selected values are included)	[OrderAmount] Between 100.00 And 1000.00
In	Equal to any of several values	[State] In ("TN", "CA", "PA")

the Abreon group

Operator	Description	Criteria Example
Is	With other operators to test conditions	[State] Is Null, [State] Is Not Null
Like	To compare two strings (can use wildcards)	[State] Like "T*"
Null	Field value is blank (empty). Can be combined with **Not** to find all non-blanks.	Is Null Is Not Null

Tips:

- Access is very forgiving about the syntax used in the query criteria. Access tries to determine the exact syntax and will modify the contents of the criteria section to match its exact syntax. If a problem occurs, please review the "new" criteria after running a query to see if the syntax was interpreted correctly.

- Pressing **[Shift] + [F2]** will display the **Zoom** dialog box which shows the entire currently selected expression from the QBE grid.

- For more information on conditional operators as criteria, use the Answer Wizard in Access Help. Search for the topic "Examples of Expressions" and then display the "Examples of expressions used in queries and filters". Select the "Examples of criteria used to retrieve records" subtopic.

Using Joins in Queries

Joins between tables are usually established in the **Relationships** window when the tables are designed. However, additional relationships can be created in the Query Design window, or the properties of a relationship can be changed for a particular query.

A selected field can be dragged to a field in another table to create a join between the two tables. These fields do not need to be primary keys to create the relationship and there is some flexibility with respect to the join that is created.

Procedure:

1. After the tables have been placed in the **Query Design** window, select the field in the first table.

2. Drag this field to the desired join field in the second table and release the mouse button. A line between the fields will be created showing the **Join** relationship.
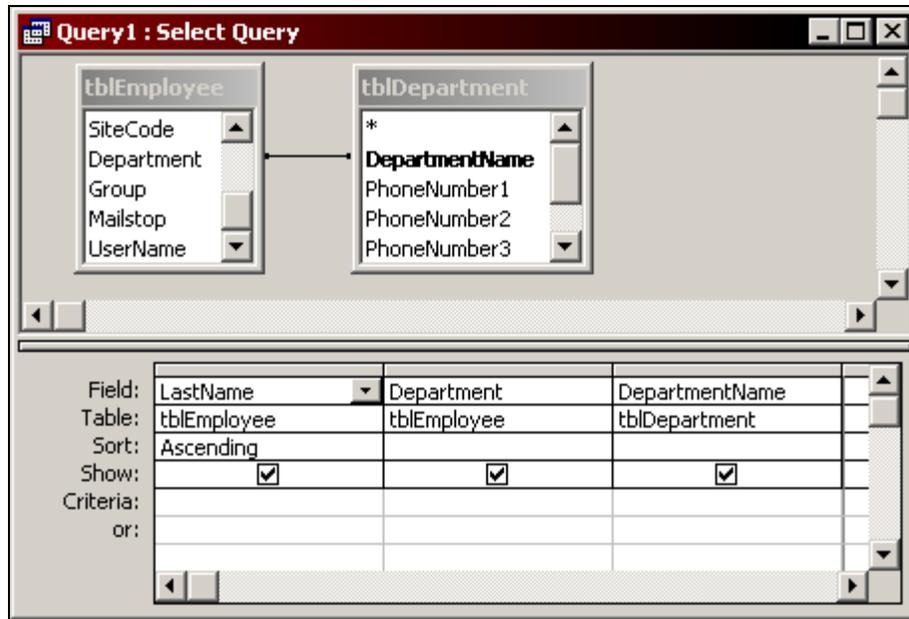

3. *Double*-click the join line to change the properties of the join. The **Join Properties** dialog box displays:
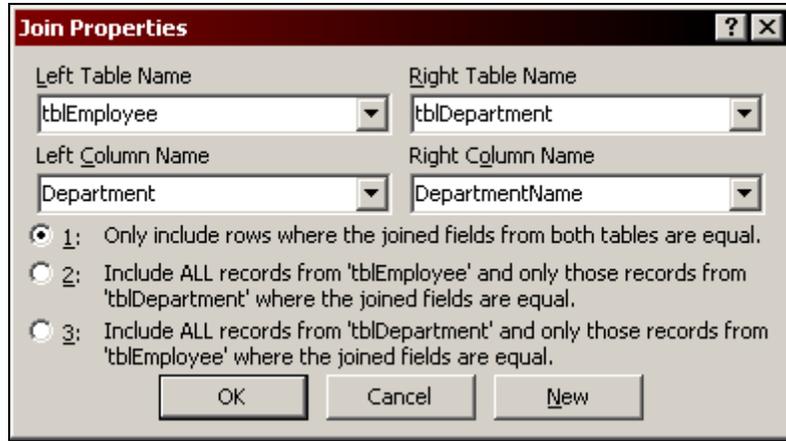
4. The following table explains these options and what types of results they will produce:

Option	Type	Description
1	**Inner Join**	Only includes records from BOTH tables where the fields have equal values. In this example, the query returns 22 records. Only records where the Department name field matches in both tables (tblEmployee and tblDeparment) are displayed.

Option	Type	Description
2	**Right Outer Join**	Used for showing all records in the left table while showing common records in the right table. The table or query that does not have a matching record will simply display empty cells in the datasheet.
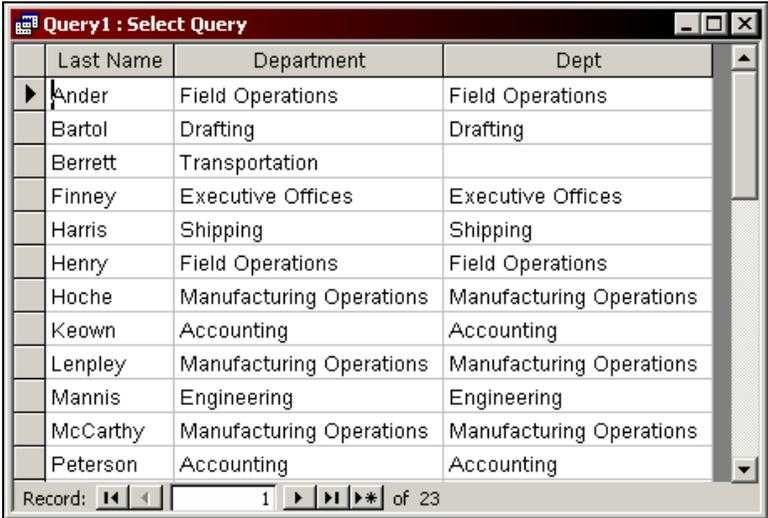
In this example, the query returns 23 records. Notice that employee **Berrett**, who was not in the inner join, is included in these results. Berrett's department as listed in tblEmployee – Transportation – does not have a corresponding record in tblDepartment.

Option	Type	Description
3	**Left Outer Join**	Used for showing all records in the right table while showing common records in the left table. The table or query that does not have a matching record will simply display empty cells in the datasheet.
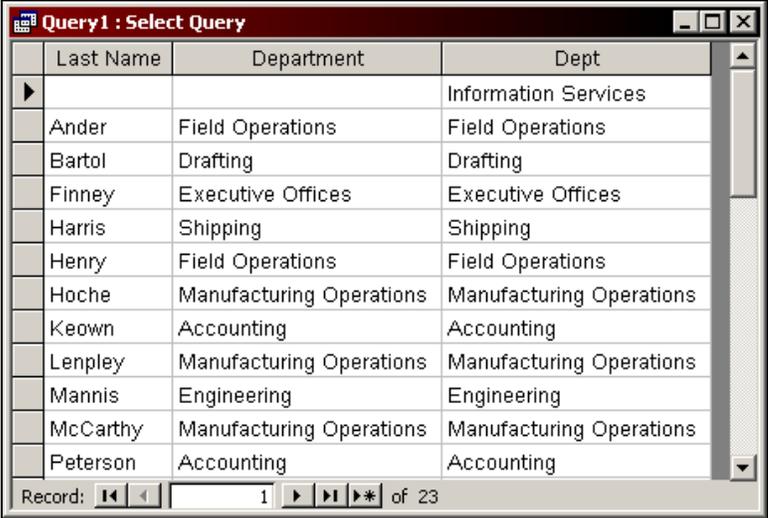
This query also returns 23 records. In this case, employee Barrett does not display, however, the department "Information Services" now appears at the top of the list with blanks in the LastName and Department fields. This means that no employees in tblEmployee currently have a department of Information Services, but the value exists in tblDepartment.

5. Select the desired join type and click **OK**.

6.   Click the **Run** button on the **Query Design** toolbar to execute the query and view the results.

Tips:

- This join exists only in the query in which it was created. Creating another query with this join requires recreating the join. The join will be saved with this query and used every time the query is run.

- The join can be deleted by clicking on the join line and pressing the **[Delete]** key.

- Joins can be created to many tables and fields at once.

- If you fail to create a join line between two or more tables in a query that are not already related, the resulting dynaset will be a Cartesian product in which each and every record is joined with each and every record in the second table. If both tables are very large, it can take a substantial amount of time to produce the dynaset.

Using Parameters Queries to Set Criteria

For queries that are run often, the process of changing the criteria can be automated by creating a **Parameter Query**. When a parameter query is run, Access prompts the user for the criteria to be used in the query. Multiple parameters for a field can be specified.

Parameter queries reduce the amount of work for the database designer and user, because they eliminate the need to open the query in Design view and change the criteria each time before they are run.

In beginning Access design, the **Enter Parameter Value** dialog box is often encountered by accident, usually when a user or database designer misspells the name of a field name in the QBE grid.
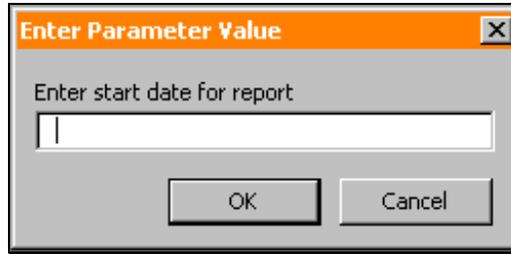
Procedure:

1. Display the query in **Design** view.

2. In the **Criteria** cell for the field, type the message to prompt the user for the criteria enclosed in square brackets.

   Example: **[Enter start date for report]**
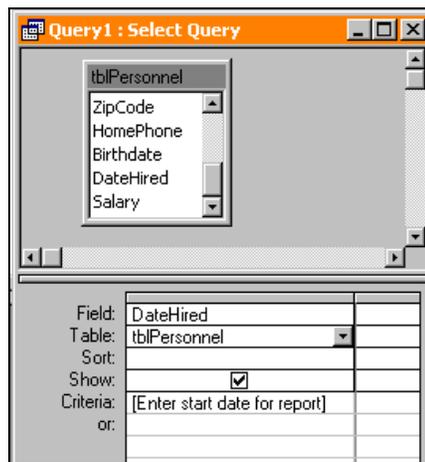
3. **Run** the query to view the results. The **Enter Parameter Value** dialog box displays and prompts for the criteria for the query:

4.   Enter the criteria in the appropriate format and click **OK**.

Tips:

- There is no character limit to the size of the parameter criteria, however, since the **Enter Parameter Value** dialog box is non-modal (i.e. its size cannot be changed), only the first part of the expression may be displayed if there is too much text. A fairly good guideline is to keep the parameter criteria to less than 32 characters.

- Multiple parameter prompts can be set up for the fields a in query, as well as a single criteria may contain multiple parameter prompts.

  For example:

  Between [Enter start date] and [Enter end date]

Working with the Expression Builder

The **Expression Builder** accelerates the process of creating expressions anywhere that you would normally need to create one, such as default values, calculated fields, and criteria in queries.
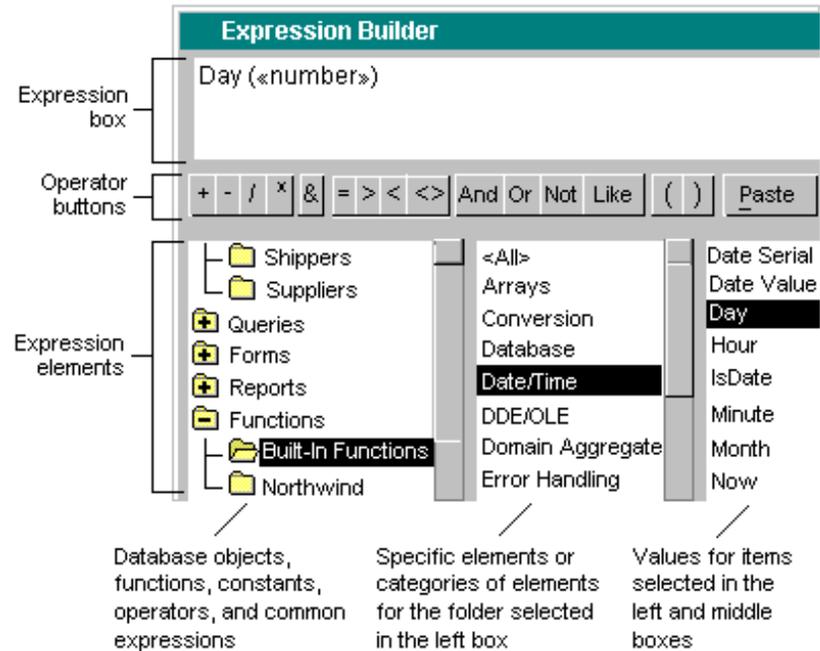
In the upper section of the builder is an expression box where you build the expression. Use the lower section of the builder to create elements of the expression, and then paste these elements into the expression box to form an expression. You can also type parts of the expression directly into the expression box.

In the middle section of the builder are buttons for commonly used operators. If you click one of the operator buttons, the Expression Builder inserts the operator at the insertion point in the expression box. For a complete list of operators you can use in expressions, click the **Operators** folder in the lower-left box and the appropriate operator category in the middle box. The right box lists all operators in the selected category.

In the lower section of the builder are three boxes:

- The left box contains folders that list the table, query, form, and report database objects, built-in and user-defined functions, constants, operators, and common expressions.

- The middle box lists specific elements or categories of elements for the folder selected in the left box. For example, if you click **Built-In Functions** in the left box, the middle box lists categories of Microsoft Access functions.

- The right box lists the values, if any, for the elements you select in the left and middle boxes. For example, if you click **Built-In Functions** in the left box and a category of functions in the middle box, the right box lists all built-in functions in the selected category.
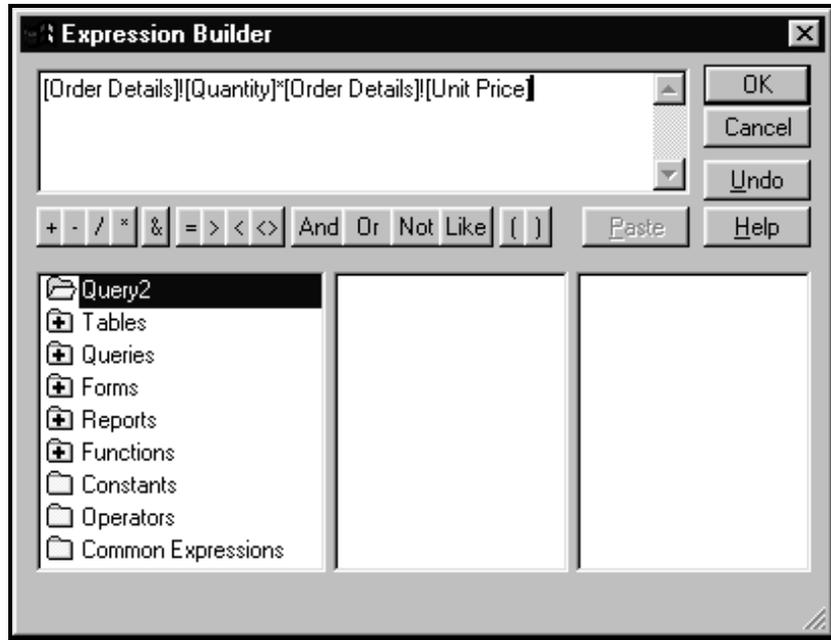
Procedure:

1. Display the query in **Design** view.

2. Click in the **Criteria** line of the field in which the expression is to be built.

3. Click the **Build** button on the **Query Design** toolbar. The **Expression Builder** dialog box displays:
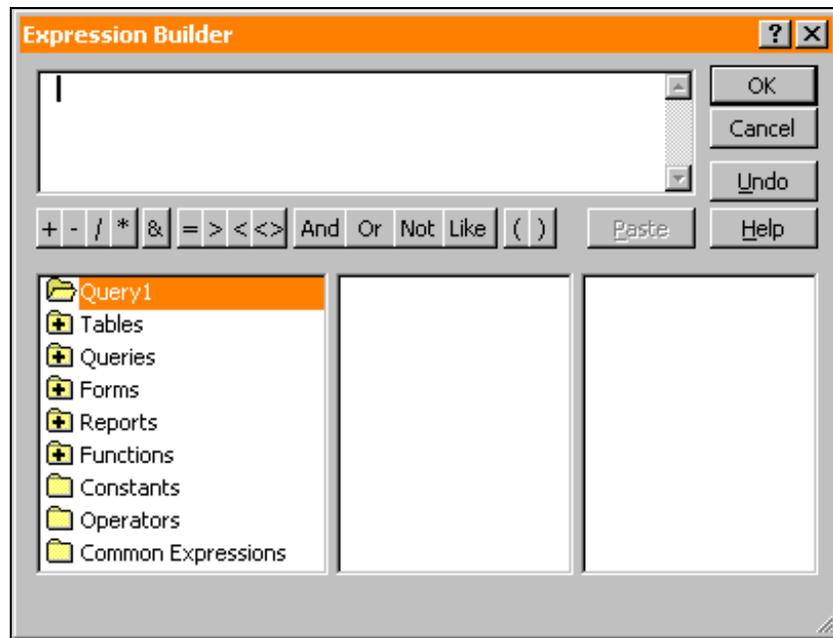
4. Select or input the fields, values, and operators which you wish to enter.

5.   Click **OK**.

6.   The expression will be pasted in the criteria cell where the Expression
     Builder was started from.

Modifying Query Properties

Like all Access objects, queries have many properties associated with them that can be changed by the user. These properties can affect how and when the query is run, as well as what types of information are output into the resulting dynaset.

You can use the query-level properties just as you would the properties in forms, tables, and reports. The query level properties available depend on the type of query being created, and on the table or field with which you're working.

Procedure: (To view query properties)

1. Display the query in **Design** view.

2. Click a blank spot in the table list area.
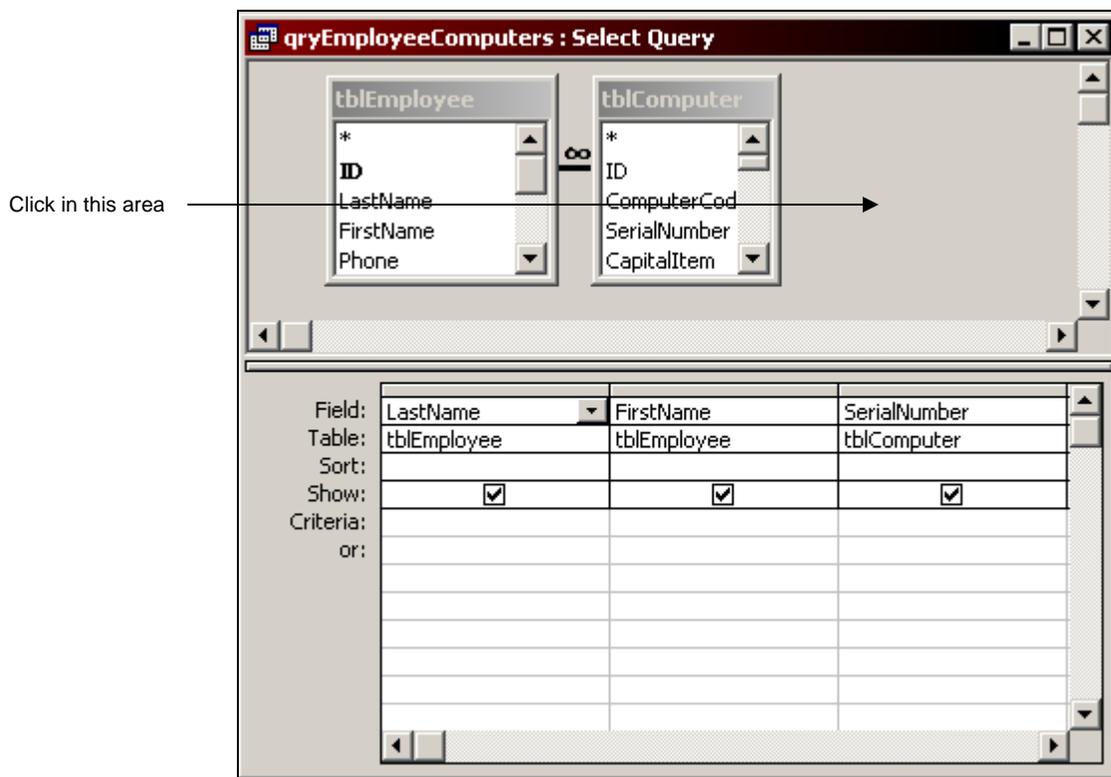
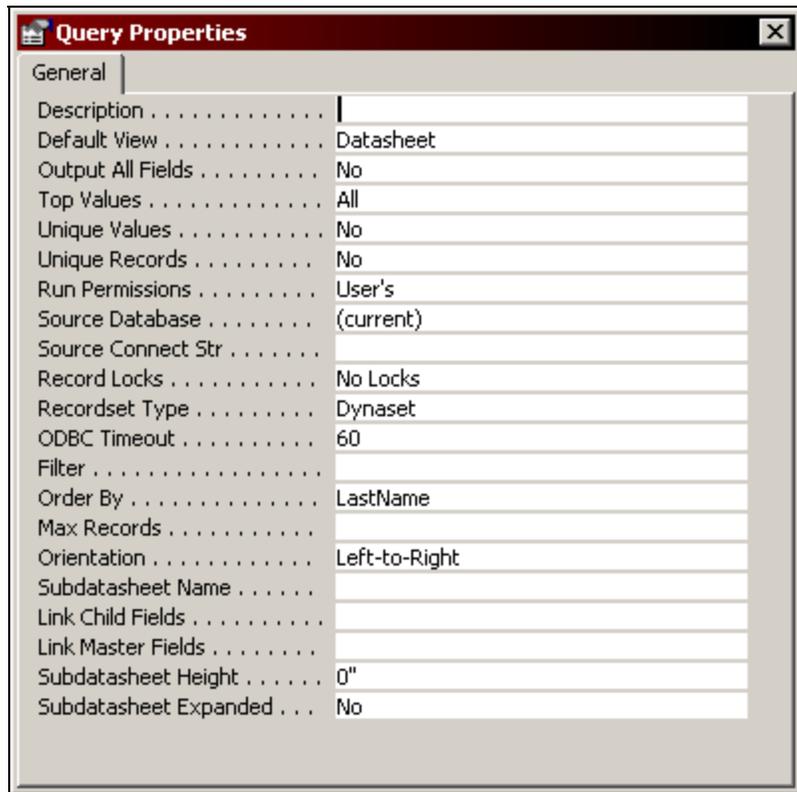

3. Select the **View** menu and choose **Properties**.

OR

Click the **Properties** button on the **Query Design** toolbar.

<p align="center">**OR**</p>

Double-click anywhere in the query design window

<p align="center">**OR**</p>

Press **[F4]**. The **Query Properties** dialog box displays:

```
Query Properties                                    ×
┌ General ┐
Description . . . . . . . . . . . . . |
Default View . . . . . . . . . . . . Datasheet
Output All Fields . . . . . . . . . No
Top Values . . . . . . . . . . . . . All
Unique Values . . . . . . . . . . . No
Unique Records . . . . . . . . . No
Run Permissions . . . . . . . . . User's
Source Database . . . . . . . . (current)
Source Connect Str . . . . . . .
Record Locks . . . . . . . . . . . No Locks
Recordset Type . . . . . . . . . Dynaset
ODBC Timeout . . . . . . . . . . 60
Filter . . . . . . . . . . . . . . . . .
Order By . . . . . . . . . . . . . . LastName
Max Records . . . . . . . . . . .
Orientation . . . . . . . . . . . . Left-to-Right
Subdatasheet Name . . . . . .
Link Child Fields . . . . . . . . . .
Link Master Fields . . . . . . . .
Subdatasheet Height . . . . . . 0"
Subdatasheet Expanded . . . No
```

4.   The following table explains these options:

Property	Description
Description	A text label describing the table or query.
Default View	The default view of the query when it is run.
Output All Fields	Choice of "yes" or "no" to show all fields from the underlying tables in the query.
Top Values	Number or percent of rows the query will return in the dynaset.
Unique Values	Choice of "yes" or "no" to return only unique field values in the dynaset.
Unique Records	Choice of "yes" or "no" to return only unique records for the dynaset. This is useful for queries used as the basis for combo or list boxes. This should not be selected with crosstab queries.
Run Permissions	Establishes whether permissions should be considered when running the query. This option allows or restricts users who are not permitted to view the data in the underlying tables to view the query dynaset or run an action query.
Source Database	Name of the current database that is the source of the query.
Source Connect Str	The name of the application used to create an external database.
Record Locks	In a multiuser database, specifies how records in an underlying table or query are locked while query runs (usually in action queries), or when two users try to edit the same record at the same time.
Recordset type	Specifies what kind of record set is available to a form.
ODBC Timeout	Time in seconds to cancel ODBC query if server or database doesn't respond.
Filter	Specifies a subset of records to be displayed when a filter is applied to a query.
Order By	Sort order loaded automatically with the query.

Property	Description
Max Records	The maximum number of records the query can return from an ODBC source.
Orientation	Specifies the view orientation. (Primarily used for languages that read right-to-left instead of left-to-right.)
Subdatasheet Name	Determines the table or query that is bound to the subdatasheet.
Link Child Fields	The field to be used as the foreign key or matching field.
Link Master Fields	The primary key or matching field for the subdatasheet.
Subdatasheet Height	The default height to display in the subdatasheet
Subdatasheet Expanded	Specifies the saved state of subdatasheets – whether they display as expanded or collapsed.

5.   Make any changes as desired to the query properties.

6.   Click the **X** in the upper right-hand corner to close the **Query Properties** dialog box.

Chapter 3

Using Query Wizards

Chapter Objectives:

- Creating a Crosstab Query

- Finding Duplicate Records

- Finding Records without Matches

Overview

Although queries can be created directly in design view, Access offers a shortcut using the Query Wizard. By answering a few simple questions, the Query Wizard creates the basic query design to which the user can make customized modifications. This utility greatly simplifies the query designing process.
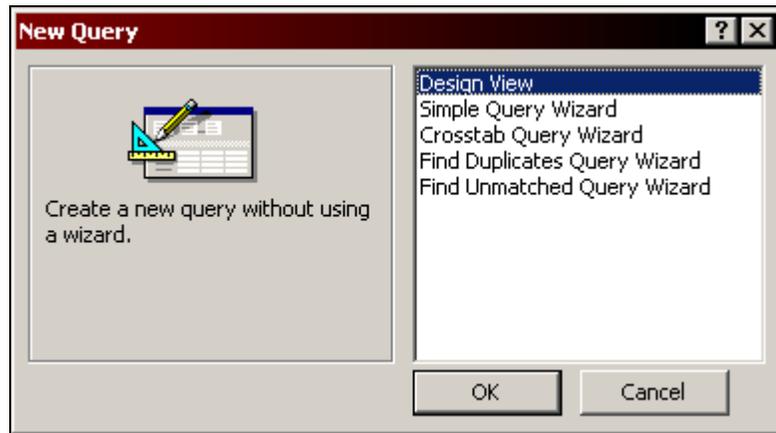
Access furnishes four different wizards from which to choose:

- **Simple** – Creates a simple select query based on fields you choose.

- **Crosstab** – Creates a query which organizes your data into unique groups based on records with common entries in a given field. (Similar to a PivotTable in Excel.)

- **Find Duplicates** – Designs a query which shows the number of appearances of a given value in the underlying table or query.

- **Find Unmatched** – Provides a dynaset which shows all of the records in one table which do not appear in a related table.

This chapter will explore the use of the latter three query wizards and how they can be useful in an Access database.

Creating a Crosstab Query

This type of query is similar to a spreadsheet calculation of **X** and **Y** values. It has the grouping features of a group totals query (to be seen in Chapter 4), but also adds an additional row to the QBE grid to establish the following pieces:
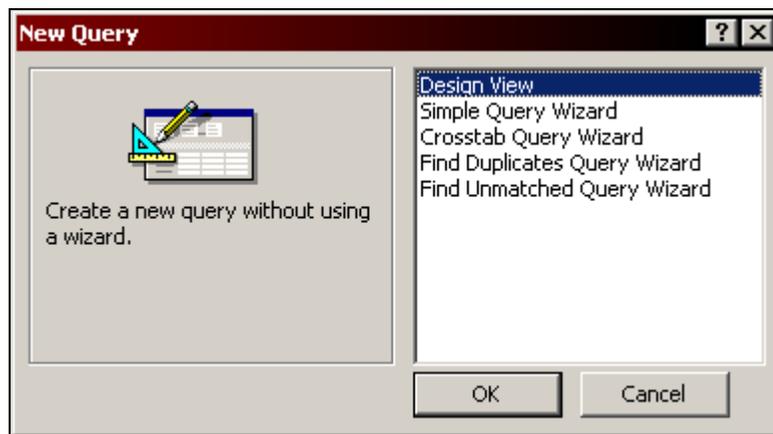
- Row Heading
- Column Heading
- Value

While crosstab queries are handy for summarizing information, you cannot sort the results based on the calculated column.

Procedure:

1. Select the **Query** tab in the **Database** window.

2. Click the **New** button. The **New Query** dialog box displays:

3. Choose the **Crosstab Query Wizard** and click **OK**. The **Crosstab Query Wizard** will begin.

4. Select the table or query as the basis for the crosstab and click **Next**.

5. Select the desired field(s) for the Row Headings from the **Available Fields** list. Use the **>** and **<** buttons to move fields to the **Selected Fields** list. Click **Next**.

6.   Select the desired field as your Column Heading and click **Next**.

7.   In the **Fields** list, select the field for which values should be computed. Depending on the field type chosen (i.e. number, text, date), the list of **Functions** in the right-hand window will vary. Click **Next**.

8.   Type a name for the crosstab query. (See **Appendix A** for guidelines on naming queries.)
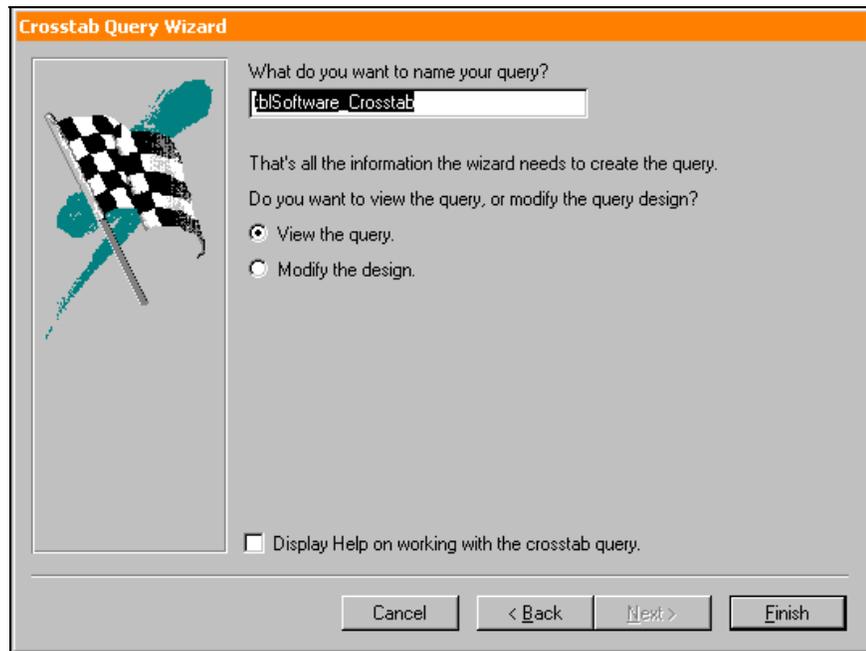
9.   Click **Finish**. The results of the query display in Datasheet view.

Tip:

- The Crosstab Query Wizard works only with one table or query at a time. To create a crosstab query based on more than one table, you must first create a query with the fields from the tables, and then create the crosstab query based on that query.

Finding Duplicate Records

The **Find Duplicates Query Wizard** can be useful when trying to locate records in a single table or query that have the same value in a field.

Lets say for example that you are trying to set a field as a primary key. When saving the table, you receive the following error:

Since duplicate values are not allowed on a primary key field, you will need to find the duplicates and change them so that they have unique values. In a small table this task would be relatively easy. However, if the table had thousands of records, this task would be much more difficult.

Using the Find Duplicates Query Wizard, you can locate the duplicates and make the necessary changes to the table data.

Procedure:

1. Select the **Query** tab in the **Database** window.

2. Click the **New** button. The **New Query** dialog box displays:
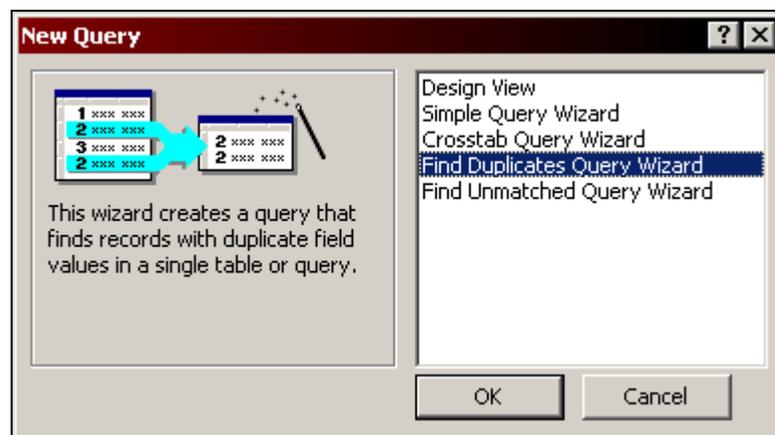
3. Choose the **Find Duplicates Query Wizard** and click **OK**. The wizard will begin.

4. Select the table or query from the list in which you wish to search for duplicate values and click **Next**.
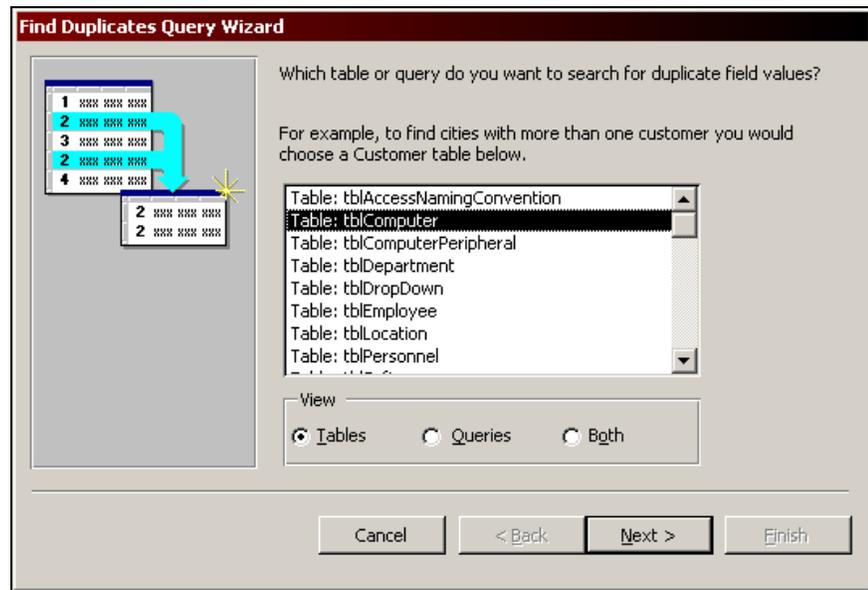
5. Select the field or fields from the **Available Fields** list that might contain duplicate information. Use the **>** button to move the field(s) to the **Duplicate-value fields** list. Click **Next**.
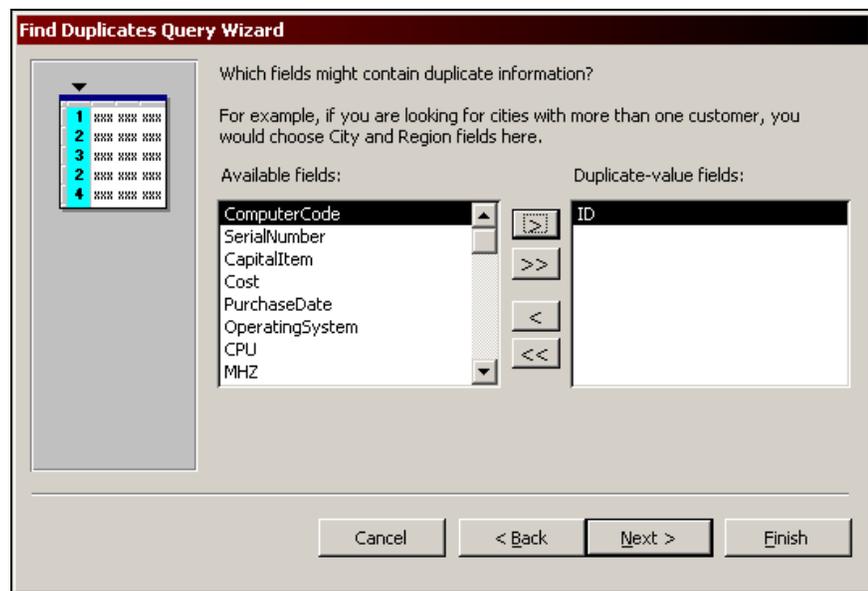
6. Select any other fields that should be displayed in the resulting query and click **Next**. You should add as many fields as necessary to help identify the records in the dynaset.
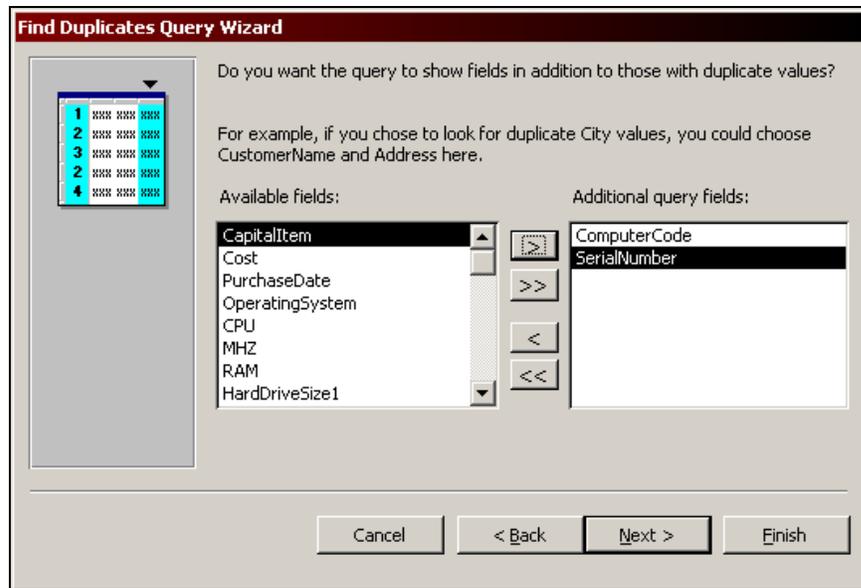
7. Name the query and click **Finish**

8.   The query results are displayed in **Datasheet** view.

9.   Since all queries display live data from their underlying tables, you can (if desired) edit the duplicate values to make them unique.

Tip:

- Once duplicate data has been changed, re-run this query again. If the resulting dynaset has no records, you have successfully eliminated the all of the duplicate values.

Finding Records without Matches

The **Find Unmatched Query Wizard** can be used to locate records in one table that do not have a match in another table.

For instance, you are working in the **Relationships** window and you try to create a relationship between two tables and enforce referential integrity when the following error occurs:
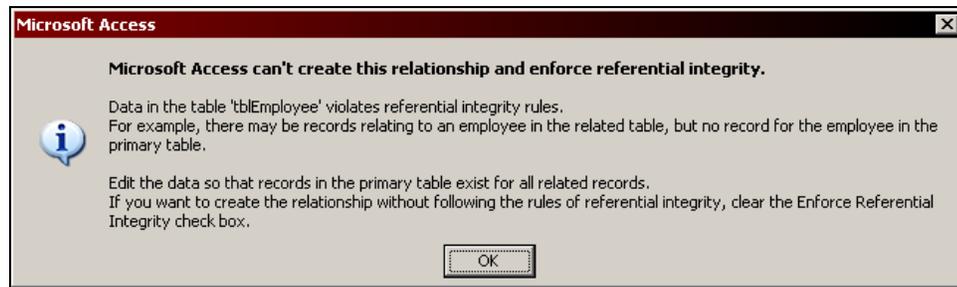
Access cannot enforce referential integrity on the relationship because one of the tables has values that do not occur in the other table. If the tables were small, you could simply do a visual examination of the tables and locate the unmatched value, however, if the either of the tables are large, this task would be much more difficult to do manually.

Using the Find Unmatched Query Wizard, you can locate the values that do not have match in a corresponding table.

 Procedure:

1. Select the **Query** tab in the **Database** window.

2. Click the **New** button. The **New Query** dialog box displays:

the Abreon group
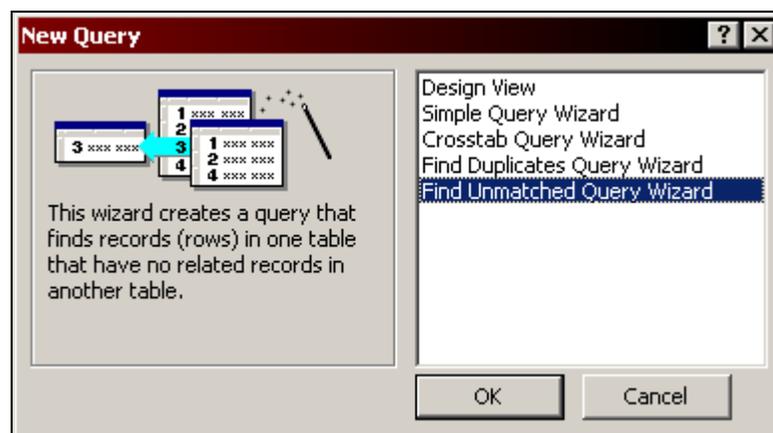
3. Choose the **Find Unmatched Query Wizard** and click **OK**. The wizard will begin.

4. Select the first table or query from the list in which you wish to search for unmatched values and click **Next**.
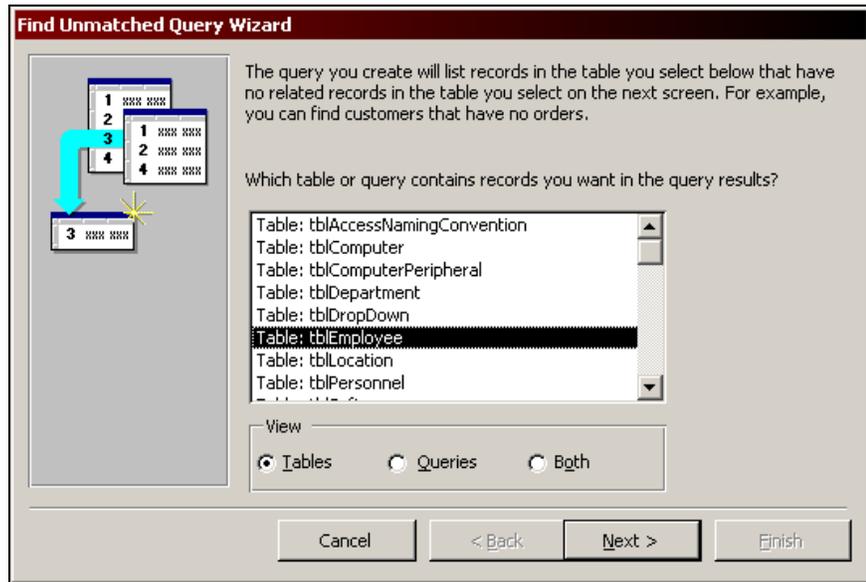

5. Select the table or query that has the related records from the first table or query you selected.

6.   In each field list, select the field that both tables share that you are trying to match up and click the **<=>** button. The **Matching Fields** section will display the selected match.
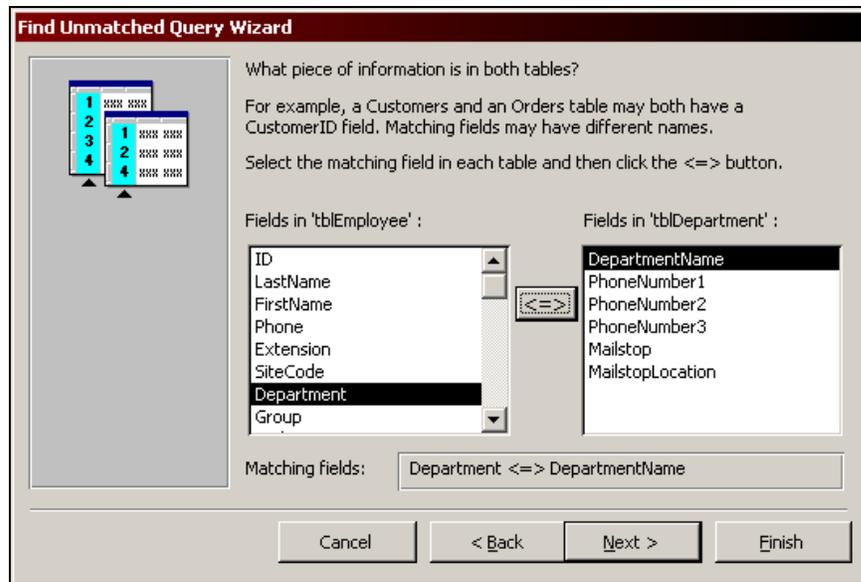
7.   Using the **Available fields** list, select any other fields you wish to have appear in the resulting dynaset. Use the **<** and **>** buttons to move fields between lists.
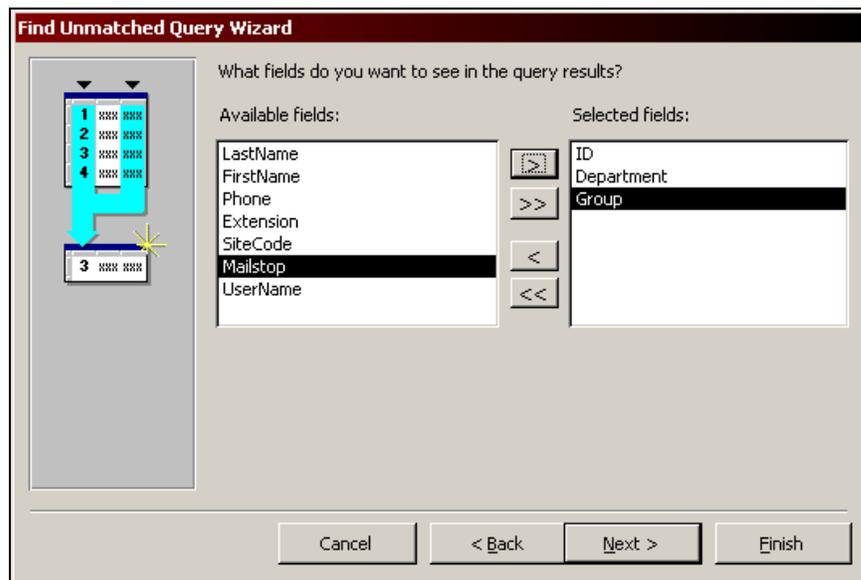
8.   Type a name for the query and click **Finish**.

9.   The results of the query display in **Datasheet** view.

10.   You can now use this information to either remove the unmatched record, or create a matching record in the joined table.

Chapter 4

Creating Calculations

Chapter Objectives:

- Adding Calculated Fields to a Query

- Summarizing Records Using a Group Totals Calculation

Overview

Not only can queries return data from tables or other queries, but they can also perform calculations on the data included in the dynaset. Having the query perform the calculation saves space in a table, reduces the amount of data-entry, and also ensures accurate calculations.

There are two main methods of including calculations in a query. The first is to create a calculated field. This method is also known as creating a horizontal calculation. The second method is to have Access summarize a group of records using a calculation. This is also known as creating a vertical calculation in a query.

Adding Calculated Fields to a Query

Access enables the user to easily perform arithmetic calculations horizontally within the query. That is, perform a calculation on fields within the same record.

For example, if a record contained the fields [Price] and [DiscountPercent], a calculated field could solve for [DiscountAmount] by multiplying the two fields to create a new one.

DiscountAmount: [Price]*[DiscountPercent]

When the calculated expression is added to the QBE grid, the dynaset will include the result of the expression in this field. No underlying data will be modified.

The following are some guidelines for creating horizontal calculations in a query:

- The name of the calculated field is the first part of the expression before the colon.

- Field names must appear in square brackets [ ].

- The expression is not case sensitive.

- The available operators are " + " (addition), " - " (subtraction), " * " (multiplication), and " / " (division).

Procedure: (To create a horizontal calculation)

1. Create a new query with the desired tables in **Design** view.

2. Add the required fields from the **Field List** box to the QBE grid.

3. Click in a **Field** line of an empty column of the QBE grid. Type the expression to be evaluated and press **[Enter]** or **[Tab]**.

4. **Run** the query to view the results in datasheet view.

Tips:

- Calculated fields can also be created using the **Expression Builder** as described in **Chapter 2 – Working with Queries**.

- Calculated fields may be used to concatenate several fields and other characters into a single field in a query. Use ampersands (&) to join the field(s) and character(s) together. For instance:

  FullName: [LastName]&", "&[FirstName]

Would produce the following results:

Last Name	First Name	FullName
Ander	Ron	Ander, Ron
Bartol	Dave	Bartol, Dave
Berrett	Jim	Berrett, Jim
Finney	Tina	Finney, Tina
Harris	Eric	Harris, Eric
Henry	Neil	Henry, Neil
Hoche	Dutch	Hoche, Dutch
Keown	Martin	Keown, Martin
Lenpley	Buddy	Lenpley, Buddy
Mannis	Ralph	Mannis, Ralph
McCarthy	Bud	McCarthy, Bud
Peterson	Lou	Peterson, Lou
Schult	Ali	Schult, Ali

Record: 1 of 23

The **FullName** column is a calculated field based on the LastName and FirstName fields.

Chapter 4 – Creating Calculations

Summarizing Records Using a Group Totals Calculation

Access can calculate group totals as well. This involves calculations to the same field(s) found throughout a group of records. Group totals are sometimes referred to as **vertical** calculations. Calculations will only be based on those records appearing in the dynaset, not the underlying table or query.

A **Group Totals** query is a specific type of select query and is accessed by selecting the **Totals** button on the **Query Design** toolbar. Selecting this button adds the **Total** line to the QBE grid. This line allows text-related fields to be grouped and number fields to be calculated.

The calculations which are available for fields include: Sum, Average, Minimum, Maximum, Count, Standard Deviation, Variance, First, and Last. These are visible in the **Total** drop-down list. Note that not every function can be used on every type of field. For instance, the Sum function cannot be used on a text field.

The **Total** line has been added to the QBE grid.
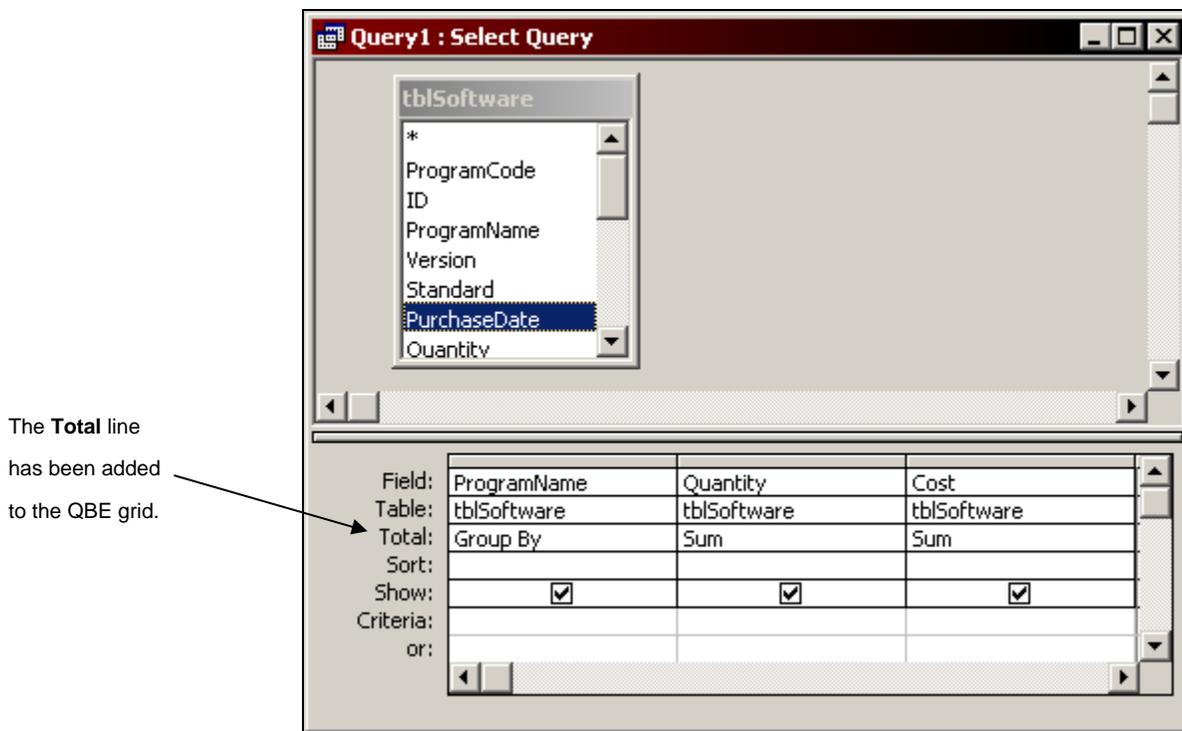
To specify criteria that is not part of the grouped field, use the **Where** option from the **Total** drop-down list.

the Abreon group

Procedure:

1. Create a new query with the desired tables in **Design** view.

2. Σ Click the **Totals** button on the **Query Design** toolbar. The **Total** line will be added to the QBE grid.

3. In the first column, select a field to be the **Group By** field. This will create the groups of data.

4. Add any additional fields as desired, selecting a function to be performed from the drop-down list in the **Total** row – Sum, Avg, Min, Max, Count, StDev, Var, First, or Last.

5. Click **Run** on the **Query Design** toolbar to run the query.

Name	SumOfQuantity	SumOfCost
ADOBE TYPE MANAGER	6	$59.00
AFTERDARK	2	$26.00
AUTO SKETCH	9	$334.00
AUTOCAD UPGRADE	1	$500.00
BORLAND C++	12	$1,042.00
BREIF	10	$89.00
C-CLEARLY	3	$125.00
CC:MAIL WINDOWS PLATFORM	4	$343.00
CENTRAL POINT PCTOOLS	5	$109.00
CHECK IT	5	$77.00
CV PERSONAL DISIGNER	3	$45.00
dBASE III PLUS	6	$443.00
DBASE IV	15	$582.00
DESIGNER/MICROGRAPHX	1	$452.00
DISKLOCK	11	$112.00
DOS 6.0	6	$10.00
DOUBLEDISK	9	$59.00
FULFILL WORD	3	$13.00
Harvard Graphics	11	$436.00
HARVARD GRAPHICS/WINDOWS	7	$399.00

Record: 1 of 93

Tips:

- Additional **Group By** fields may be defined in a Totals query. It is recommended that all Group By fields appear on the left side of the QBE in the left to right order that they performed.

- Total queries may have criteria in any **Group By** or function field. To specify criteria that is not part of the group, add the field to the end of the QBE grid and select the **Where** option.

- Fields with the **Where** option cannot appear in the resulting dynaset.

Chapter 5

Working with Action Queries

Chapter Objectives:

- Creating Tables with a Make-Table Query

- Changing Data with an Update Query

- Adding Records to a Table with an Append Query

- Deleting Records with a Delete Query

Overview

An action query is a query that makes changes to many records in just one operation. Action queries modify the information in specified tables based upon type of action query selected, and the information provided as part of the query.

There are four types of action queries:

Make-Table Query   Creates a new table from all or part of the data in one or more tables based on the criteria set by the designer.

Update Query   The Update Query will update the information in the underlying table based upon the information placed in the **Update To:** row for each field specified in the query.

Append Query   The Append Query will append the fields listed in the query to the table and fields specified as part of the append query selection process.

Delete Query   The Delete Query will delete the records from the specified tables based upon the criteria selected as part of the delete query.

You can preview a Make-Table, Append, or Delete query before it is actually executed by clicking on the **Datasheet View** button instead of the **Run** button. This allows you to make changes to the query design before the action is performed.

You cannot preview the changes that will be made to data in a table by an Update query. Viewing an Update query in Datasheet view will simply display the records that will be changed, but it will not display the actual changes that will be made.

By default, Access will display a confirmation dialog box before an action query is run. To check this setting, select the **Tools** menu, and then **Options**. In the Options dialog box, select the **Edit/Find** tab. In the **Confirm** section, make sure there is a check in the box next to **Action Queries**.

Creating Tables with a Make-Table Query

A **Make-Table** query is an action query that creates a new table based on existing data. The new table can be created within the current database, or it can be saved to a different database.

This query will make the structure for the table as well as place the data inside the table, however, the new table won't contain the original table's primary keys, indexes, or column and table properties (other than the defaults assigned to all tables).

Procedure:

1. Create a new query in **Design** view which results in the desired dynaset, including any criteria as necessary to limit the results.

2. Select the **Query** menu and choose **Make-Table Query**.

OR

Click the **Query Type** button on the **Query Design** toolbar and choose **Make-Table Query** from the list. The **Make Table** dialog box displays:
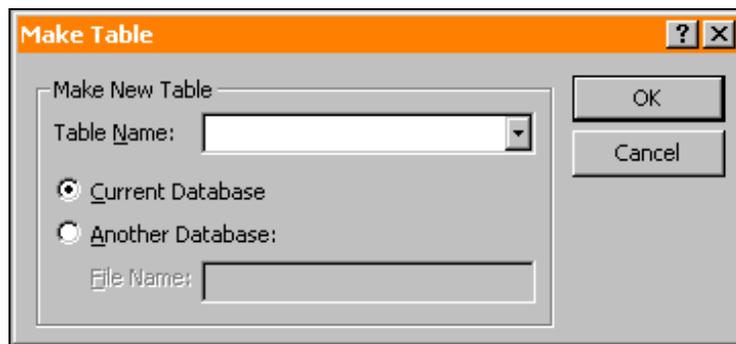
3. Enter the new **Table Name** and click **OK**.

4. If desired, click the **Datasheet View** button to preview the records that will be copied to the new table. You must switch back to **Design** view to run the query.

5. **Run** the query to create the new table.

6. A confirmation dialog box displays indicating how many records will be added to the specified table:

> **Microsoft Access** ✕
>
> ⚠ **You are about to paste 35 row(s) into a new table.**
>
> Once you click Yes, you can't use the Undo command to reverse the changes. Are you sure you want to create a new table with the selected records?
>
> [Yes]   [No]

7. Click **Yes** to execute the query.

8. **Close** and **Save** the query if desired.

Tips:

- The fields in the newly created table will inherit the data type and field size of the fields in the underlying table(s), but will not inherit any other properties.

- If the make-table query is run more than once, Access will ask to confirm if the new table should overwrite the old table. Click **No** to abort this process.

> **Microsoft Access** ✕
>
> ⚠ **The existing table 'tblSoftwareMicrosoft' will be deleted before you run the query.**
>
> Do you want to continue anyway?
>
> [Yes]   [No]

Changing Data with an Update Query

Update queries make global changes to a group of records in one or more tables, without having to make changes manually. Criteria can be specified that limit what records get modified.

For example, prices can be raised by 10 percent for all dairy products, or salaries can be raised by 5 percent for the people within a certain job category.

Procedure:

1. Create a new query in **Design** view which results in the desired dynaset, including any criteria as necessary to limit the results.

2. Select the **Query** menu and choose **Update Query**.
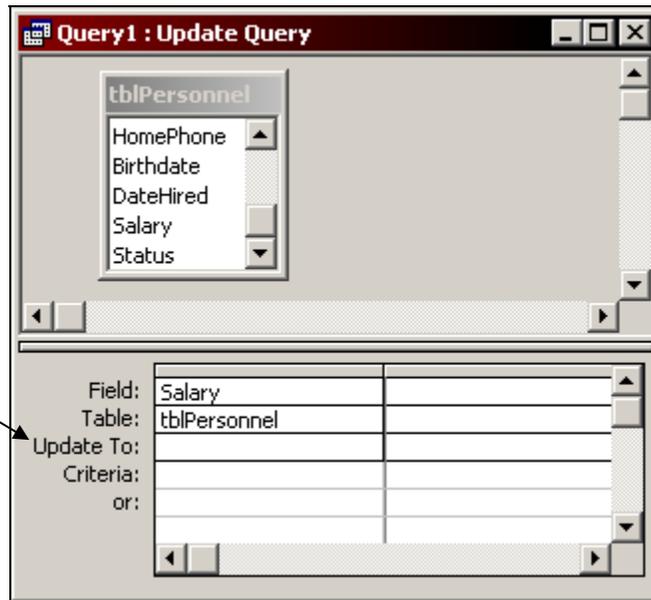
OR

Click the **Query Type** button on the **Query Design** toolbar and choose **Update Query** from the list.
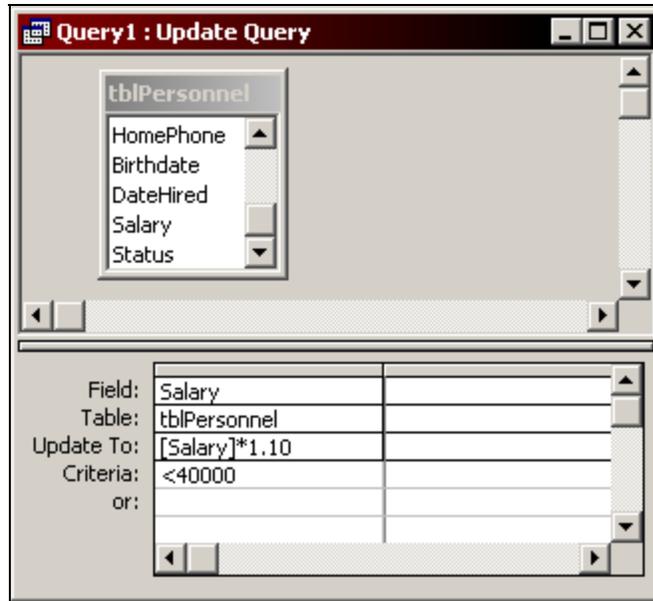
3. The **Update To** line will appear in the QBE grid.

The **Update To** line has been added to the QBE grid

4. In the **Update To** cell for the field(s) to be updated, type the expression or value that should be performed. The Expression Builder can be used to create the expression.

5. ![Run icon] **Run** the query to modify the data in the fields. Access displays a confirmation dialog box:
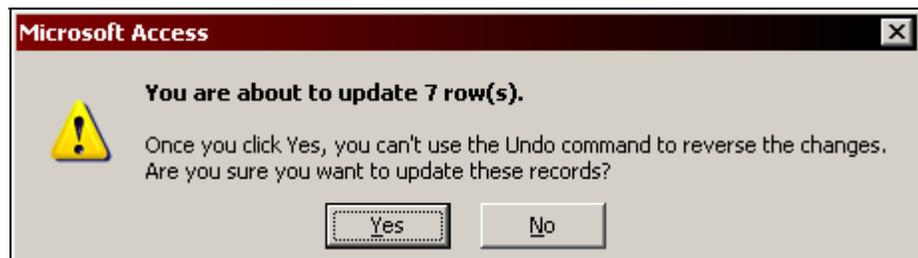
6. Click **Yes** to perform the action.

7. **Close** and **Save** the query if desired.

![Warning icon] **Warning:**

- Since it is not possible to preview the final changes made by an update query or undo the changes once they have been made, it is recommended that you perform the update query on a copy of the table before you execute it.

Adding Records to a Table with an Append Query

Append queries add a group of records from one or more tables to the end of one or more tables. For example, suppose a few new customers have been acquired and a database contains a table of information on those customers. To avoid typing all this information, you can append it to your Customers table.

Append queries can also append fields based on criteria. For example, you might want to append only the names and addresses of customers with outstanding orders.

Append queries are also good for when information is imported from an alternate source such as an Excel spreadsheet. The Excel data can first be imported to a new table, examined, fixed if necessary, and then appended to an existing table's data.

Procedure:

1. Create a new query in **Design** view which results in the desired dynaset, including any criteria as necessary to limit the results.

2. Select the **Query** menu and choose **Append Query**.

OR

Click the **Query Type** button on the **Query Design** toolbar and choose **Append Query** from the list. The **Append** dialog box displays:

Append		? X
Append To		OK
Table Name:	[ ] ▼	Cancel
● Current Database		
○ Another Database:		
File Name: [ ]		
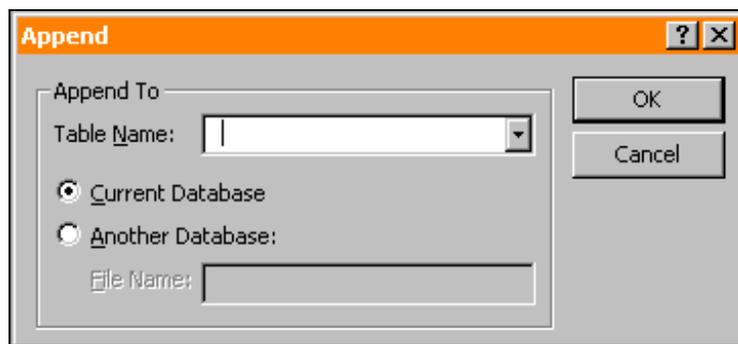
3. Select the name of the table the data should be appended to from the **Table Name** drop-down list and then click **Ok**.

the Abreon group

4.   The **Append To** row is added to the QBE grid.

The **Append To** line has been added to the QBE grid
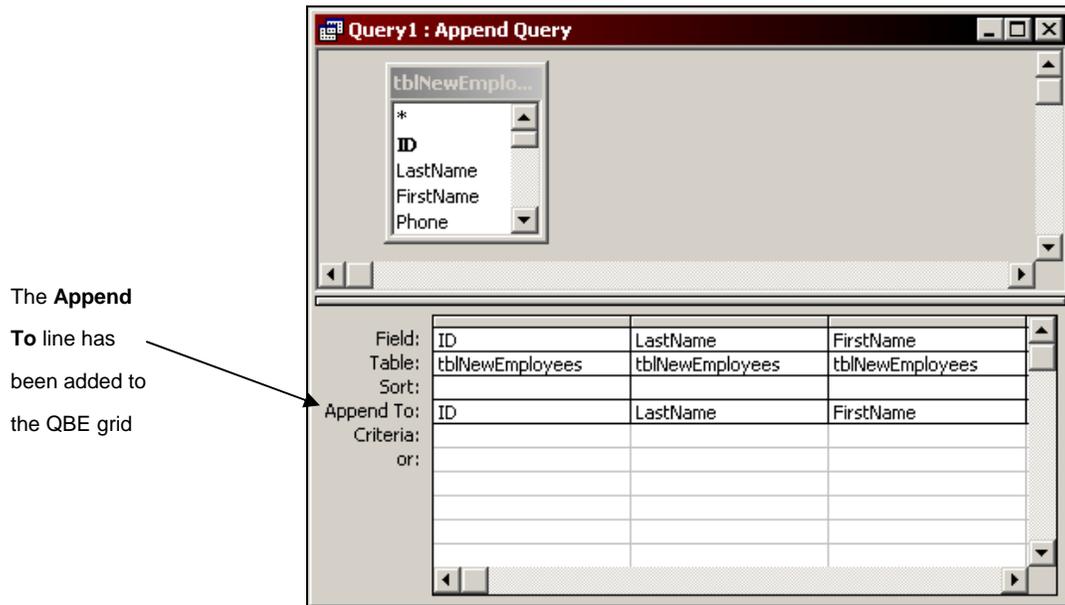
5.   Use the drop-down lists in the **Append To** row to match the query fields to the appended table. Access will automatically fill in any matching names. The field names do not need to match in the source and target tables, however, they should have the same data types.
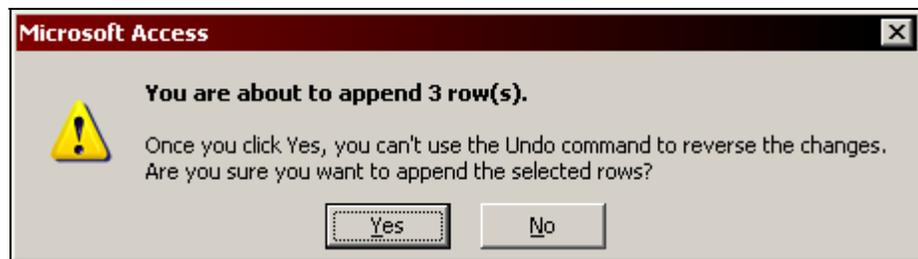
6.   If desired, click the **Datasheet View** button to preview the records that will be appended to the selected table. You must switch back to **Design** view to run the query.

7.   **Run** the query to add the records to the specified table. Access will ask you to confirm this action:

8.   **Close** and **Save** the query if desired.

Deleting Records with a Delete Query

A **delete query** deletes a group of records from one or more tables based on the criteria established by the user. For example, a delete query can be used to remove products that are discontinued and for which there are no orders.

With delete queries, entire records are deleted, not just selected fields within records.
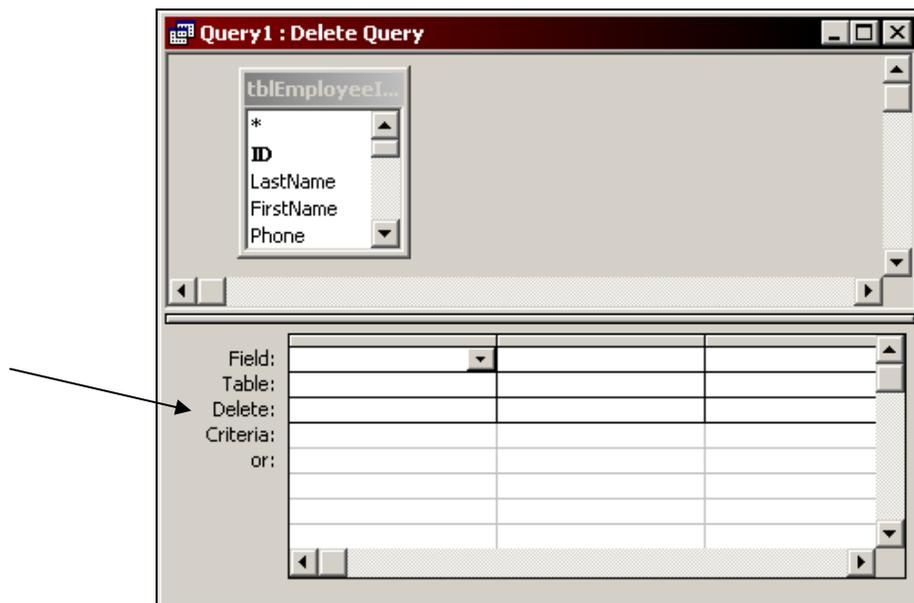
Procedure:

1. Create a new query in **Design** view.

2. Select the **Query** menu and choose **Delete Query**.

OR

Click the **Query Type** button on the **Query Design** toolbar and choose **Delete Query** from the list.

3. The **Delete** line will appear in the QBE grid.

4. (Optional) Drag the asterisk field (*) to the first column of the QBE grid. If there is more than one table in the query, this should be from the primary table of the query. **From** will appear in the **Delete** cell under this field. (This step does not affect the query – it simply allows the user to see the all of the fields from the record(s) that will be deleted when previewing or running the query.)
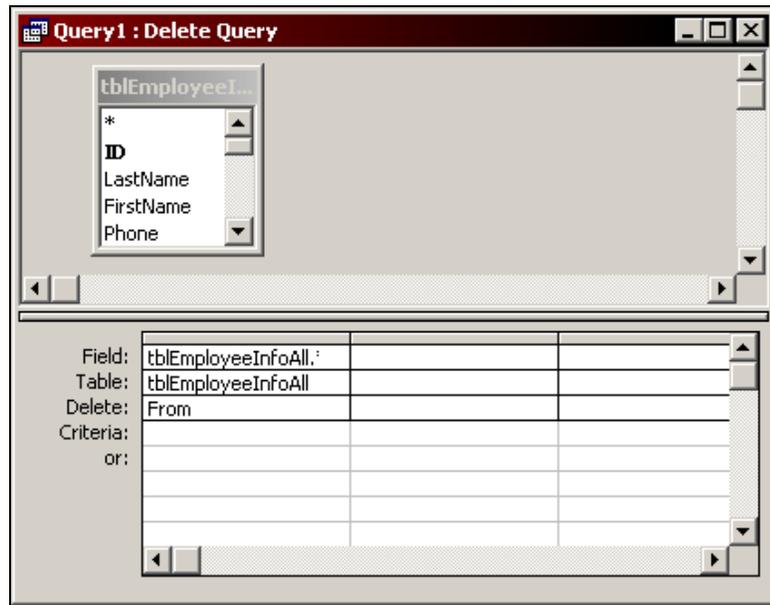
5. To specify criteria for deleting records, drag the fields on which you want to set criteria to the design grid. **Where** appears in the **Delete** cell under this field. Enter the criteria in the **Criteria** line of the field(s).

6. ⊞ ▾ If desired, click the **Datasheet View** button to preview the records that will be deleted from the selected table. You must switch back to **Design** view to run the query.

7. ⚠ **Run** the query to delete the records from the specified table. Access will ask you to confirm this action:

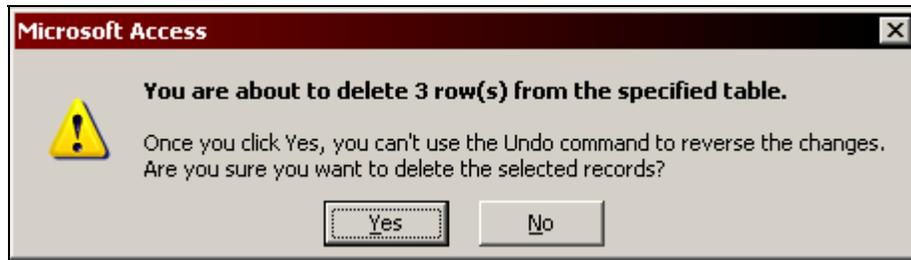| Microsoft Access	✕
⚠ **You are about to delete 3 row(s) from the specified table.** Once you click Yes, you can't use the Undo command to reverse the changes. Are you sure you want to delete the selected records?	
	[Yes] [No]

8. **Close** and **Save** the query if desired.

Tips:

- You can use a single delete query to delete records from a single table or from multiple tables in an one-to-one relationship.

- If the **Cascade Delete** option is enabled for a relationship between tables, you can use a single delete query to delete records from multiple tables in a one-to-many relationship. Otherwise, two queries must be run since a query cannot delete records from the primary table and the related tables at the same time.

Warning:

- Since a delete query cannot be undone, it is a good idea to keep a backup copy of the tables affected or the entire database in case you make a mistake or need to revert back to a previous version.

Appendix A

LNC Database Object Tags

The Leszynski Naming Convention

Many Access versions ago, two Access developers (Stan Leszynski and Greg Reddick) developed a naming convention for Access objects. The Leszynski-Reddick Naming Convention (L/R) was proposed as a standard to make it easier for developers to understand their own code (especially when returning to it after a significant amount of time had passed), and to understand databases and code developed by others.

L/R was introduced in February 1993, and was widely adopted by Access developers. Since then this naming convention has split into two branches, the Leszynski Naming Convention (LNC) and the Reddick VBA Naming Convention (RVBA), although there's considerable overlap between the two.

Using a naming convention for database objects and controls makes your database self-documenting. Every time you see a drop-down list of database objects, each item's tag will tell you what kind of object it is.

The following table lists some of the most relevant object tags that can be used by "everyday" Access users and developers.

Database Object	Tag	Example	Notes
Table	tbl	tblCustomer	
Table (lookup)	tlkp	tlkpShipper	
Query (any type)	qry	qryCustomer	
Query (append)	qapp	qappNewCustomers	
Query (crosstab)	qxtb	qxtbSalesByRegion	
Query (delete)	qdel	qdelOldAccounts	
Query (lookup)	qlkp	qlkpActiveEmployees	
Query (make table)	qmak	qmakNewFormat	
Query (select)	qsel	qselSoftwareList	
Query (totals)	qtot	qtotSoftwarePrices	
Query (update)	qupd	qupdRevisedPricing	

Appendix A – LNC Database Object Tags

Database Object	Tag	Example	Notes
Form	frm	frmDataEntrySoftware	
Form (dialog)	fdlg	fdlgDelConfirm	
Form (menu)	fmnu	fmnuPrintOptions	
Form (message)	fmsg	fmsgWait	
Form (subform)	fsub	fsubSoftwareWhoHas	
Report	rpt	rptSoftwarePurchase	
Report (subreport)	rsub	rsubSoftwareSummary	
Data access page	dap	dapOnlineEntry	
Macro	mcr	mcrOpenCriteriaForm	
Module	bas	basCustomFunction	
Any Object	zz	zztblOldSoftwareList	Any old object no longer in use, but wanted for possible reuse, reference, or backup.
Any Object	zt	zttblNewSoftwareFormat	Programmatically created temporary object.
Any Object	zs	zsrptDatabaseActivity	Displayed system objects.
Any Object	_	_tblSoftwareList	Under development, incomplete, or backup copy. Remove when complete, don't reference anywhere until "_" is removed.

the Abreon group

Appendix B

Query Toolbars

Query Design View Toolbar

The following table describes the toolbar buttons available when in the **Design View** of a query.

Button	Name	Function
	View	Click once to switch to Datasheet View, or click the drop-down arrow to select a view from the list.
	Save	Saves the query.
	Search	Opens the Search task pane.
	Print	Not available in Design view.
	Print Preview	Not available in Design view.
	Spelling	Not available in Design view.
	Cut	Removes a selection and places it on the clipboard.
	Copy	Copies the selected item to the clipboard.
	Paste	Pastes a copied or cut item from the clipboard to the selected location.
	Undo	Removes a previous action.
	Redo	Redoes a previous action.
	Query Type	Click the drop-down arrow to display a list of query types.
	Run	Executes the query.
	Show Table	Opens the Show Table dialog box.

Button	Name	Function
Σ	**Totals**	Adds the Total line to the QBE grid.
All ▾	**Top Values**	Changes the properties of the query to display the top values as selected from the list.
	Properties	Displays the Properties dialog box for the selected item.
	Build	Opens the Expression Builder dialog box.
	Database Window	Displays the main database window.
▾	**New Object**	Click to create an AutoForm based on the selected object, or click the drop-down arrow to display a list of objects to be created.
?	**Help**	Opens the help system.

Query Datasheet View Toolbar

The following table describes the toolbar buttons available when in the **Datasheet View** of a query.

Button	Name	Function
	View	Click once to switch to Design View, or click the drop-down arrow to select a view from the list.
	Save	Saves the query.
	Search	Opens the Search task pane.
	Print	Prints the query results in datasheet format to the default printer.
	Print Preview	Displays a preview of the printed output.
	Spelling	Starts the spell checker.
	Cut	Removes a selection and places it on the clipboard.
	Copy	Copies the selected item to the clipboard.
	Paste	Pastes a copied or cut item from the clipboard to the selected location.
	Undo	Removes a previous action.
	Insert Hyperlink	Inserts a hyperlink at the selected location. Can only be used on fields with a hyperlink data type.
	Sort Ascending	Sorts the records in ascending order based on the selected column
	Sort Descending	Sorts the records in descending order

Button	Name	Function
		based on the selected column
	Filter by Selection	Filters the records in the dynaset based on the selected value.
	Filter by Form	Displays a datasheet form to define criteria to limit the results in a dynaset.
	Apply Filter	Applies or removes a selected filter.
	Find	Opens the Find and Replace dialog box to search for values or records.
	New Record	Moves the cursor to the end of the dynaset to a new record.
	Delete Record	Deletes the current record. Displays a confirmation dialog box before executing this action.
	Database Window	Displays the main database window.
	New Object	Click to create an AutoForm based on the selected object, or click the drop-down arrow to display a list of objects to be created.
	Help	Opens the help system.

Index

S

T

U

V

W

Z

the Abreon group