Access Queries
(Office 2010)

Application Support and Training

Office of Information Technology, West Virginia University

OIT Help Desk – (304) 293-4444

oit.wvu.edu/training/classmat/db/

Instructors: Kathy Fletcher, Bob Haring-Smith, Roman Olynyk, Steve Marra

Last revised: January 24, 2011

Table of Contents

Course Description

This is the third in a series of six workshops about Microsoft Access. It deals specifically with queries, plus some information about filters.

Many databases contain large, varied amounts of information. There are times when you will need to narrow the focus of the information in the database or combine information from multiple tables. This is done through the use of queries. Queries allow you to view, change, and analyze data in different ways. You can also use them as a source of records for forms and reports.

The purpose of this session is to familiarize you with some of the different types of queries and how they can be used. Our goal is to assist you to learn the software, understand some basic concepts and show you some tips and techniques so you can develop your database management/programming skills over time.

The six Access workshops are:

- Introduction to Access
- Access Database Design
- Access Queries
- Access Reports
- Access Form Design
- Access Macros and Database Utilities

Thank you,

The OIT Technical Support Services Trainers

West Virginia University

What is a Query?

In the strictest sense, a query is a way to answer questions about the data in your tables. It allows you to retrieve items in a database that meet a set of criteria that you specify. However, within the realm of modern databases, a specialized form of a query can also be used to:

- Define database objects
- Perform calculations
- Make new tables
- Append, delete, or update records
- Produce data summaries

Access contains some other features that are not strictly classified as queries, but they can be used to achieve query-like results. These are sorting and filters. Sorting was covered in the Introduction to Access workshop. We will discuss filters today before we tackle actual queries.

Filters

A filter can be used to select or exclude from a table a set of records that meet specified criteria. We'll see in a moment how to apply filters, but first, here is some information about them.

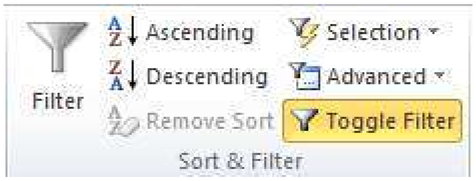
Limitations of Filters

While filters are powerful in their search and retrieval capabilities, they do have limitations.

- Selection limited to one table
- Cannot calculate new data
- Must retrieve the entire record

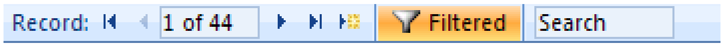
Types of Filters

This table lists types of filters and indicates where each type can be found in the Access 2010 interface. Table Header Menus can be seen by clicking on the down arrowhead at the right end of each column's header in a table's datasheet view. The filter tools are in the **Sort & Filter** group, which is in the Home ribbon. In addition to the Sort & Filter command, a **Shortcut Menu** appears when you click the right mouse button while the mouse cursor is over an appropriately chosen cell of a table.



Filter	Action	Table Header Menus	Home Ribbon	Shortcut Menu
Filter by Selection	Uses selected text as criterion	X	Selection tool	X
Filter by Exclusion	Uses all text except selected text as criterion	X	Selection tool	X
Filter By Form	Uses sample text entered into a blank record as criteria	X	Advanced tool	
Advanced Filter/Sort	Uses the Query by Example (QBE) grid to set criteria	X	Advanced tool	

Notes on Filters

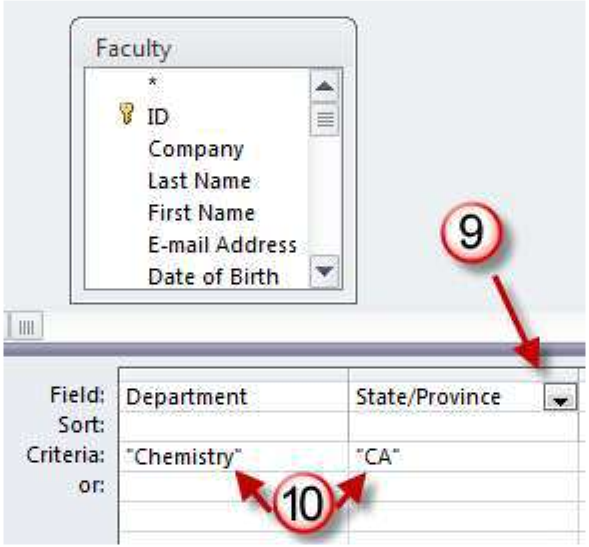
- A filter can be saved as a query. You must be in the Advanced Filter/Sort view and then choose **Advanced** **Save as Query** from the ribbon
- To remove a filter and view all records within a table, click on the **Toggle Filter** tool in the Home ribbon or click on the **Filtered** section of the status bar at the bottom of the table pane: 

- Multiple fields can be used in a filter; Filter will evaluate from left to right
- For each specified field in an Advanced Filter, you can indicate either a sort order or criteria or both

Activity 1: Creating and Using Filters

In this activity you will try out the **Filter by Selection** technique and explore the **Advanced Filter**/**Sort** tool, which allows you to save your filter as a Query.

1. Open the Faculty database and enable content.
2. Open the Faculty table.
3. Click in any Department field that has a value of “Chemistry”
4. Perform a Filter by Selection to focus on all records in which the Department is “Chemistry.” Use the Selection tool in the Sort & Filter group on the Home ribbon. However, you can also do this operation by right-clicking over your selection and using the shortcut menu, or you could click on the header menu for Department.
5. You should now see eleven records, displaying only those individuals who are listed in the Chemistry department.
6. Remove the filter by clicking on the Toggle Filter command, which is also in the Sort & Filter group on the Home ribbon. In addition, the header menu for Department also has a selection for clearing the filter.



Field:	Department	State/Province
Sort:		
Criteria:	"Chemistry"	"CA"
or:		

7. Choose Advanced Filter/Sort using the Advanced tool in the Sort & Filter group on the Home ribbon.
8. In the lower portion of the screen, Department should be listed in the field row.
9. Click in the Field textbox of the next column Using the drop down list, choose the State/Province field
10. Enter “Chemistry” in the criteria area of the Department field (it may already be there) and “CA“ in the criteria area of the State/Province field
11. Click on Apply Filter/Sort through the Advanced tool in the ribbon.   Your filter for Chemistry department faculty from California should have found one record for Dorothy Till.
12. Return to the Advanced Filter/Sort pane by clicking on the Advanced tool again and selecting Advanced Filter/Sort from the menu.
13. Remove the criterion from Department field, so that we are only filtering for state.
14. Enter “MA” and “TN” in the OR: textboxes below “CA” in the State/Province field.
15. Choose Save As Query from the menu Advanced (while you are viewing the advanced filter).
16. Save the query with the name CAMATN.
17. Apply the filter. How many customers are from one of these states? There should be eleven records, with none from MA.
18. Remove the filter
19. Close the FacultyFilter and the Faculty table.

20. Open the CAMATN query by double clicking on it where it appears in the list of queries in the Navigation pane.
21. Switch to Design View (click on the blue design icon in the Views group).
22. Close the CAMATN query

Queries

Why Use Queries?

In most cases, the real reason for using a database to store data records is so they can be retrieved using criteria that are specified interactively. Since this is the definition of a query, we could logically assume that queries are the main reason for using a database. Access gives you a great deal of flexibility and power with queries. You can use them to:

- **Choose fields:** You can select the exact fields needed to satisfy the query. You do not have to include all of the fields of a record in a query
- **Choose records:** You can select records from a table based on specified criteria
- **Sort records:** You can view the results of the query in a specified order
- **Access multiple tables:** You can select data from several different tables and place the results in a single dynaset[1]. You can even access data from other database management systems
- **Perform calculations:** You can add new fields to the dynaset containing the results of calculations involving other fields
- **Provide data source:** You can use a query as a source of data for forms, reports, and other queries
- **Update data:** You can update, delete, or append a group of records in one step. You can even create a new table that includes some or all of the records from an existing table or tables

Types of Queries

Query Type	Description
Select Query	A select query is used to view, correlate, or analyze the data in one or more tables. It is the most common type of query in Access, and it is very similar to the previously discussed filter.
Crosstab Query	A crosstab query is used to aggregate the data in categories that label the rows and columns like a pivot table in Excel. It can summarize large amounts of data in an easy-to-read format and it blends well with reports and charts.
Make-Table Query*	Creates a new table from the results of a select query. It can be used to make table backups, to archive old data,

[1] ***Dynaset*** is the official name for the table of results from running a query.

	or to extract a table of matched data for exporting.
Update Query*	Lets you make global changes to data in one or more tables. Allows you to enter one set of changes that could be posted to many records with minimal effort.
Append Query*	Allows you to add records to the end of a table. It can be used to merge tables together by appending one table to the end of another.
Delete Query*	Deletes a group of records from one or more tables. Could be used in conjunction with the Make Table query to delete archived records from a table after an archive table was created and filled.
Union Query	A union query is used to combine the matching fields from two or more tables.
Pass-Through Query	A pass-through query is used to send commands to a SQL database.
Data Definition Query	A data definition query creates, changes, or deletes tables in an Access database using SQL statements.

* These are action queries. When these queries are run design changes are made to the database in some manner.

Creating a Query

Queries can be created in either of two principal ways: the Query Wizard or the Query Design View. A tool for each is provided in the "Queries" section of the Create ribbon. Let's start by looking briefly at the Query Wizard.

Query Wizard

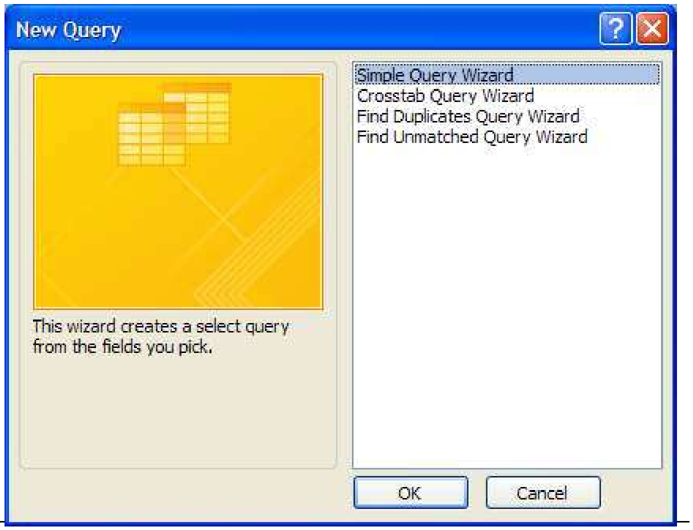
If you click on the **Query Wizard** tool in the Create ribbon, you will be offered these choices:

Simple Query Wizard
Create a select query from one or more tables for simple selections and calculations

Crosstab Query
Create a crosstab table based on a single table or previously created query

Find Duplicates Wizard
Find duplicate records within a single table or query

Find Unmatched Wizard
Find records in one table that do not



This wizard creates a select query from the fields you pick.

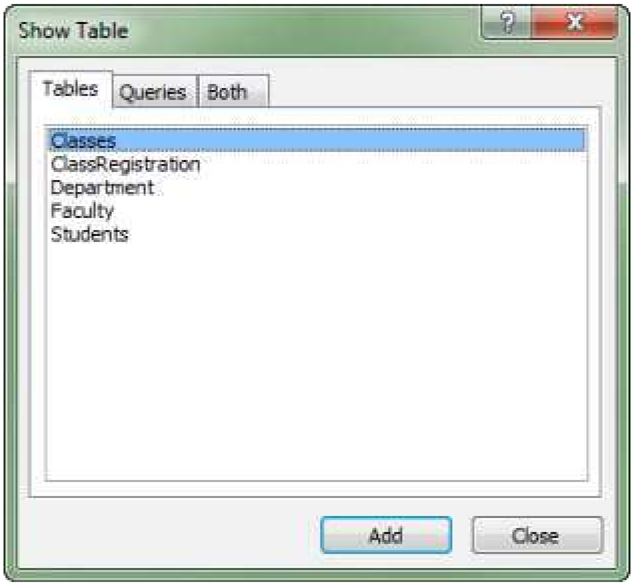
match records in a related table

Select the method that you wish to use and click on **OK**.

From this point, you'll be led through the process of choosing the tables or queries—and the fields from those tables or queries—that will feed your new query. The specifics of the process vary with the type of query that you want to create. Once the query has been created, changes must be made either by making different choices when using the wizard or by using the Query Design View to modify the query created by the wizard.

Using Design View to Create Queries

Unlike for forms and reports, using Design View from start to finish is a reasonably easy way to create and refine a query. When you first click on the **Query Design** tool, you will be presented with the Show Table window, which gives you a chance to specify the sources of data to be used in the query.



Classes
ClassRegistration
Department
Faculty
Students

Select a data source

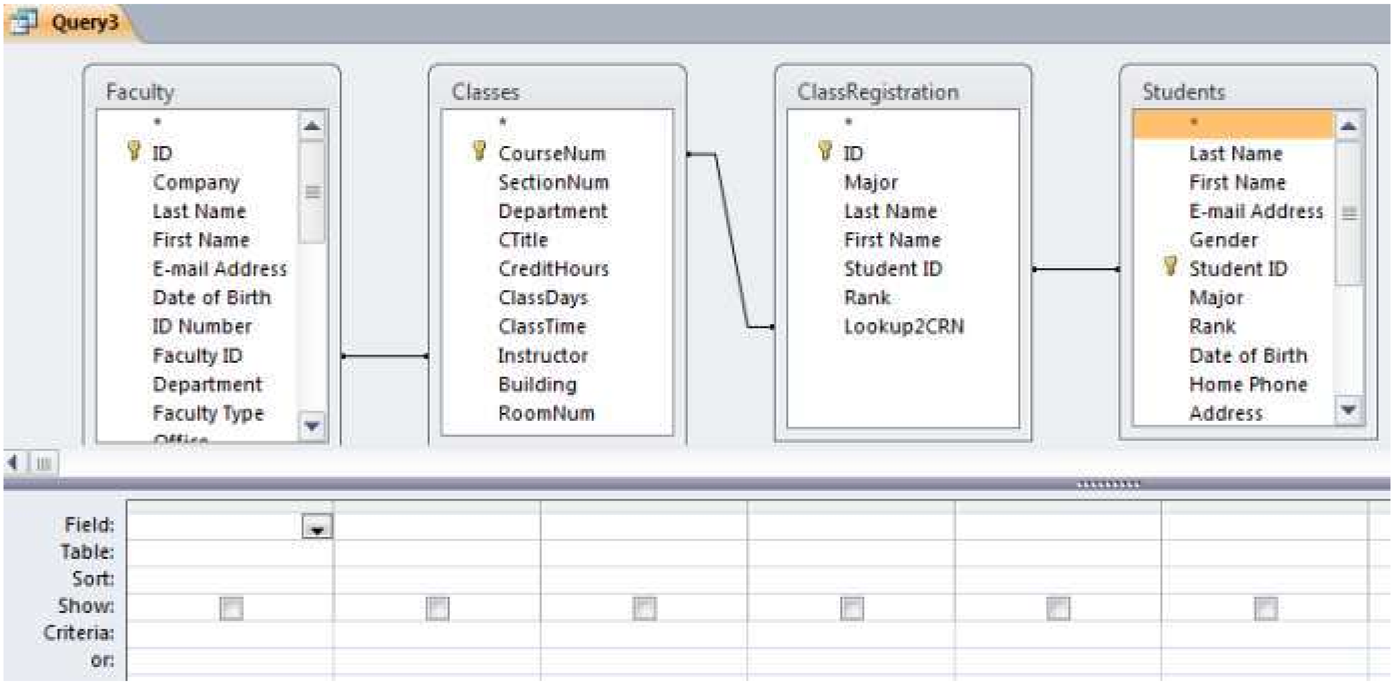
1. Click on the name of the table or query that you wish to use in your query.
2. Click on **Add**
3. Repeat for each table that you want to add to the current query
4. Click on **Close** when finished

Alternative technique: you can double click on the table names to add them to the query.

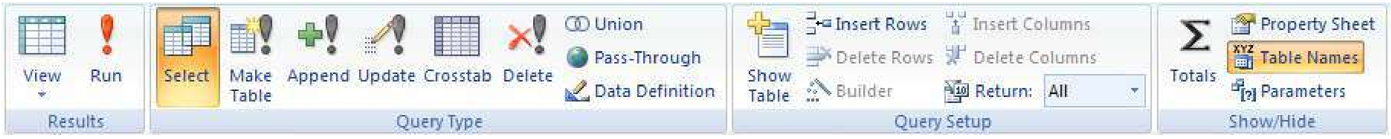
When you have finished selecting data sources, you will be looking at the "query by example" window, which is illustrated below.

Query By Example Window

Here is a query by example (QBE) pane in which the Faculty, Classes, ClassRegistration, and Students tables have been selected as data sources. Note the join lines linking a field from one table to another – these lines indicate relationships between the tables.



When viewing a QBE pane, you should expect to see the Design ribbon under Query Tools at the top of the Access window. The ribbon looks like this:



Query Notes:

- If more than one table is selected, join lines should be displayed between the tables to show the relationships (a primary key in one table is tied to a foreign key in another table).
- You can add more tables to a query at any time by choosing **Show Table** from the Design ribbon, or right-clicking in the background of the QBE pane and selecting **Show Table** from the pop-up menu
- The type of query that you are creating is indicated by the highlighted tool in the Query Type section of the Design ribbon. In the image above, the Select tool is highlighted. Most of the queries we create in this workshop will be select queries.

Selecting Fields

Only the fields that are added to the QBE grid will be displayed in the resulting dynaset. Fields can be added from multiple tables as long as there is a connection or relationship between the two tables.

Fields can be added to the Query in one of three ways:

1. In the Field row of the QBE grid, click on the drop down menu and choose the field name that you want to add. So that you can distinguish between two fields of the same name in different tables, the names are presented in the form tablename.fieldname. For example, the Company field in the Customers table would be designated Customers.Company.
2. Click and drag the field that you want to add from the table listing in the top portion of the screen to a field cell in the lower portion.

3. Double click on the field name that you want to add from the table listings; the field should then appear in the left-most unoccupied column of the QBE grid.

Use any combinations of these methods to add all of the fields to the query that you wish to include.

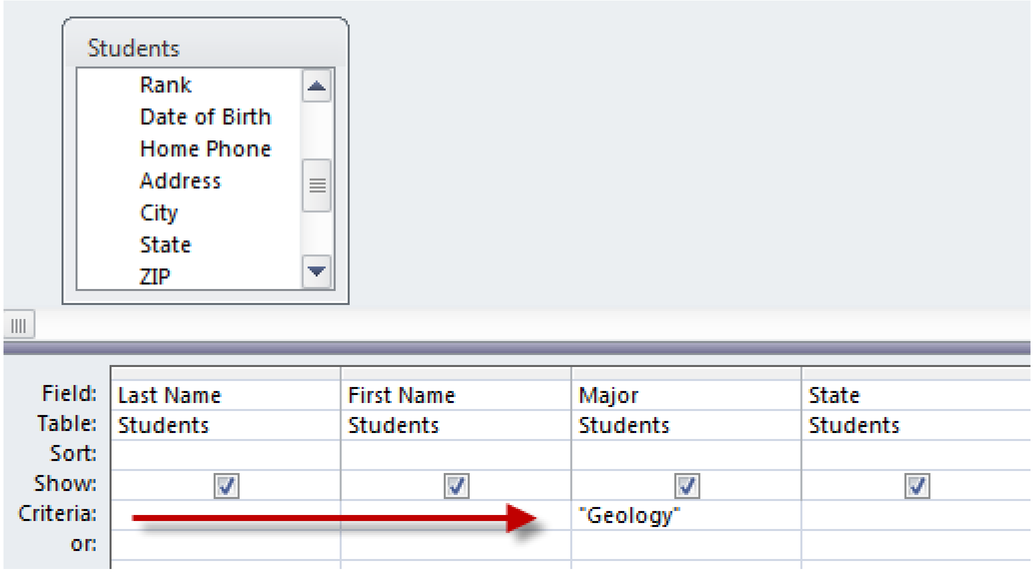
Query Views

While we will most often look at queries either in the Design View (to modify the structure of the query) or the Datasheet View (to see the data identified by the query), the drop-down menu associated with the View tool at the left end of the Design ribbon for queries offers a few other views as well. Here is the complete list:

View	Description
Design View	use the QBE grid to create or modify queries
Datasheet View	view the results of the query in a table-like array
SQL View	view or modify the SQL code associated with the query
PivotTable View	view or organize the results of a query as a PivotTable
PivotChart View	view or organize the results of a query as a PivotChart

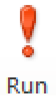
Activity 2: Create a Simple Query

1. Click on the **Create** ribbon, if it is not already selected.
2. In the Queries group, click on the **Query Design** tool
3. From the Show Table dialog box, select and add the **Students** table
4. Close the Show Table dialog box.
5. Add the fields **Last Name, First Name**, **Major**, and **State** to the QBE grid
6. Type the word “Geology” in the criteria row below **Major**



Field:	Last Name	First Name	Major	State
Table:	Students	Students	Students	Students
Sort:				
Show:	☑	☑	☑	☑
Criteria:			"Geology"	
or:				

7. Run the query by clicking on the **Run** button (signified by the large red exclamation point) or by switching to **Datasheet View.**



8. Notice that the results of a Select Query only display the selected fields, as opposed to all of the fields being displayed via the Advanced Filter/Sort that we had previously run.
9. Close the query and save it with the name **FindGeoMajors.**

Query Utilities

Rearranging, Inserting, or Deleting Fields

While you are selecting the fields to be used in the query, it's easy to alter the query structure if needed. You can re-order, delete, or insert additional fields.

Moving a field
To move a field left or right in the grid, click on the grey header bar to highlight the field. Then click on it again and drag it to a new location. **Note:** It's important to use two separate clicks; a single click and drag will simply highlight several fields.

Deleting a field
To delete a field, click on the grey header bar to highlight the field. Then press the Delete key.

Inserting a field
To insert a field, select it from a table field list and drag it to the column position where you want it inserted. The other columns will shift to the right to make room for it.

Specifying Sort Order

When you view the results of a query, it's often easier to interpret the data if it is sorted into some logical sequence. Sorts can be specified directly in the Query By Example (QBE) grid.

- Click in the Sort cell for the field that is to be the sort key, pull down the sort options menu, and select the desired sort order

- To sort on multiple fields, arrange them from left to right in the desired sort key order and select the desired sort options for each field. **Note:** In a query, you cannot independently control the sort order for a multi-field sort and the display order for those fields. You could, however, create a report based on the query in which the fields are displayed as you wish without affecting the sort order.

Excluding Fields from the Dynaset

Fields can be used as part of the selection criteria but excluded from the resulting dynaset. To exclude a field from the displayed dynaset, remove the check from the box in the Show cell of the QBE grid.

Specifying Criteria

To select certain records from the database tables, we can specify selection criteria that a record must meet in order to be selected. The criteria are specified in the form of an expression.

- A simple expression would be a value. The query would search for all records with that value in that field.

- You can use the symbols <, >, <=, >=, and <> (not equal to) along with a value. For example, a criterion of **<200** (i.e., less than 200) could be specified for a cost field.
- More complex expressions can also be created to search for a range of values or the difference between values. The expression builder can be used to create the expression that describes the criteria of the query. Start the expression builder and click on the Help button for more information on using the builder.
- If you know the syntax of the expression, you can simply type it in the criteria cell
- Wildcards can be used to specify the criteria. You can use ? in place of any single character, * in place of any number of characters, and # in place of any single numeric digit.

Combining Selection Criteria

Selection criteria entered into the Query By Example grid can be combined within columns or across columns. The logical operators AND and OR are available for use but are generally implied by the position of the criteria within the grid.

- To use a multiple OR condition within a column, list the first criteria in the cell marked Criteria and placed successive items in the cells directly below the first
- A multiple AND condition within a column is generated by placing both criteria in the same cell separated by the keyword AND
- An AND condition across columns is created by placing the specifications in cells along the same criteria row
- An OR condition across columns is created by placing the specifications in cells in different rows

Viewing the Resulting Dynaset

After the criteria specifications are entered in the Query By Example (QBE) grid and the query is executed, Access extracts the resulting data from the table and places it in the dynaset.

You can view the dynaset by choosing **Datasheet View** from the ribbon. The dynaset (sometimes referred to as a recordset) resulting from a query appears to have all of the characteristics of a table, but it isn’t quite the same. It is called a *virtual table* because it exists in the memory of the computer only. When you exit Access, the dynaset is lost unless you take some direct measures to save it. Procedures for converting the dynaset to a table are described later in this handout.

Saving the Query

When you save a query, it becomes part of the database and appears under the **Queries** entry in the **Objects** list in the Database window.

Saving a query saves only the structure of the query itself. It does not save a snapshot of the dynaset results from running the query. When you open the query in the future, it will re-run the query and display a new dynaset result.

To save a query, do one of the following:

- From Design View, right-click on the tab for the query and select **Save** from the pop-up menu.
- Close the query. If you have not yet saved changes to the query, Access will prompt you to do so when you attempt to close the query.

- Click on the File tab, and select **Save Object As**. The subsequent **Save As** dialog allows you to provide a new name for your query.

The first time that you save a query, you'll be asked to name it. It is advisable that you do not give a query the same name as one of your tables; that will make the table and the query harder to distinguish in listings that include both tables and queries. Also, if you will be creating a serious database project, do not use blanks or symbols in your query's name (other than numbers and underscores); Visual Basic for Applications modules will be easier to create if you have used simpler object names. One naming convention uses the letter "q" at the beginning of all query names.

Printing Query Results

You can print the results of the query by selecting the Datasheet View and choosing the **Print** command from the **File** ribbon. The output will be presented in table form with no formatting.

To produce a nicely formatted report, you will have to create a report. We will discuss that topic in a subsequent workshop.

Activity 3: Combining Criteria in a Single Field

Up to now, we've been making queries with single criteria, such as finding all Geology majors. In *Combining Selection Criteria* above, we learned that logical operators can be used within a column. In this activity we will search the Faculty table for individuals who were born during the decade of the 60s.

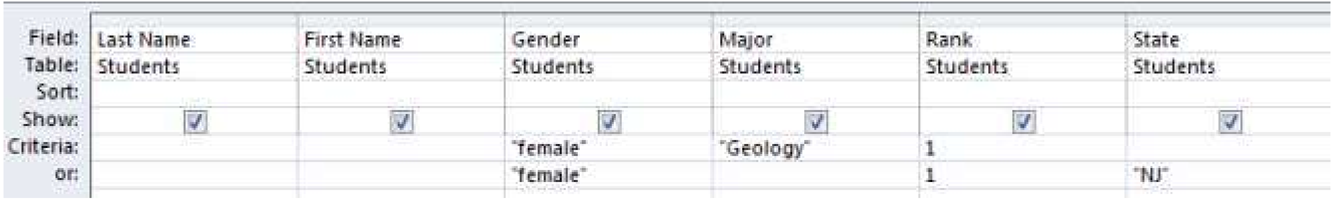
1. Click on **Query Design** in the **Create** ribbon.
2. Add the **Faculty** table
3. Close the Show Table dialog box.
4. Add the fields **Last Name, First Name, Department,** and **Date of Birth.**
5. In the criteria row beneath **Date of Birth,** type in **>= 1/1/1960 AND <= 12/31/1969**.
6. Run the query and review the results (you should have twelve people).
7. Switch back to **Design View**
8. Change the Sort row beneath **Date of Birth** to Ascending, so the birthdays will be in chronological order.
9. Run the query and review the results.
10. Close the query and save it with the name **Children of the 60s**

Activity 4: Combining Criteria using Multiple Fields

When constructing queries across multiple fields, it is important to understand the logic . A geology sorority wants a list of freshman women for a Geo-Girls mixer. To liven things up a bit, they also want to invite any freshman women from New Jersey.

1. Click on **Query Design** in the **Create** ribbon.
2. Add the **Students** table
3. Close the Show Tables dialog box.
4. Add the fields **Last Name**, **First Name**, **Gender, Major**, **Rank**, and **State** to the grid
5. Set the criteria under **Gender** as "female."

6. Set the criteria under **Major** as “Geology.”
7. Set the criteria under Rank as “1,” which denotes 1st year students or freshmen.
8. Now if we were to put “NJ” in the first criteria position under **State**, we would be limiting the search to JUST freshman female geology majors from New Jersey (one person!). And if we were to put “NJ” by itself in the OR condition for State, we would get our female freshman geology major plus ***everyone*** – regardless of gender, major, or rank -- from New Jersey. What we want to do instead is to create a duplicate OR condition for Gender and Rank, with an added “NJ” in the State position:



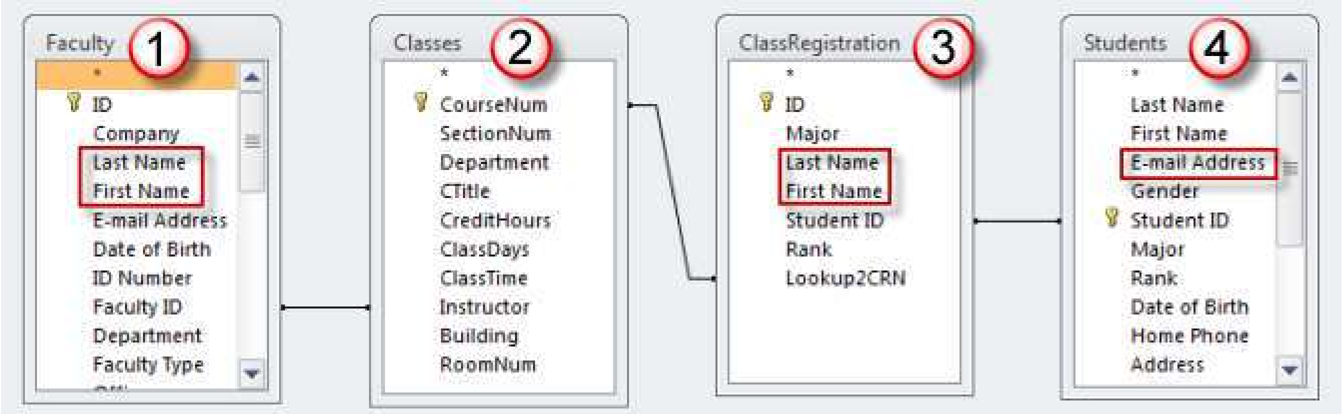
Field:	Last Name	First Name	Gender	Major	Rank	State
Table:	Students	Students	Students	Students	Students	Students
Sort:						
Show:	☑	☑	☑	☑	☑	☑
Criteria:			"female"	"Geology"	1	
or:			"female"		1	"NJ"

9. Run the query and review the results (you should have 30 “hits.”
10. Close the query and save it with the name **GeoGirlsMixer.**

Activity 5: Creating a Multi-table Query

The power of a relational database becomes apparent when you can query for records across multiple tables.

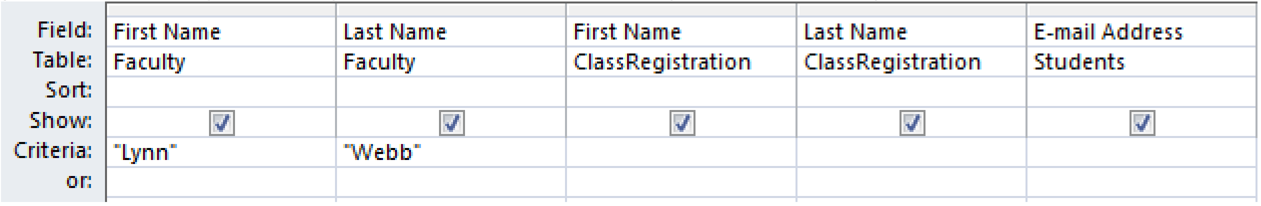
Physics professor Lynn Webb is going to retire soon. We want to send an email inviting her current students to the retirement party. In order to accomplish this, our query must incorporate multiple tables. When querying across multiple tables, however, it is important that you consider how these tables are related:



It should be obvious that we need the Faculty table (1) in order to find Lynn Webb. The ClassRegistration table (3) can provide us with the names of students who are taking a specific class, but note that there is no direct relationship between the Faculty table and the ClassRegistration table. In order to work, our query must include the Classes table (2) even though we don’t specifically need a field from it. In addition, a student’s email address is only available from the Students table (4), so we’ll also need to incorporate that table.

1. Click on **Query Design** in the **Create** ribbon.
2. Add the **Faculty**, **Classes**, **ClassRegistration**, and **Students** tables
3. Close the Show Tables dialog box.

4. Add the fields **First Name** and **Last Name** (from Faculty), **First Name** and **Last Name** (from ClassRegistration), and **E-mail Address** (from Students)



Field:	First Name	Last Name	First Name	Last Name	E-mail Address
Table:	Faculty	Faculty	ClassRegistration	ClassRegistration	Students
Sort:					
Show:	☑	☑	☑	☑	☑
Criteria:	"Lynn"	"Webb"			
or:					

5. Type “Lynn” in the Criteria row under **First Name** for Faculty.
6. Type “Webb” in the Criteria row under **Last Name** for Faculty.
7. Run the query (you should have 51 records). Note that the Faculty First Name and Last Name, although needed for the actual query, are not needed for the final result. The Show boxes can be deselected now that we know the query is working properly.
8. Close the query and save it with the name **StudentsOfWebb**

Calculations

Within Access you can perform calculations to display results in a field that are not stored in the database tables. Expressions are used within a query to define the desired calculations

- Calculations can be entered manually in the Field cell of an empty column of the QBE grid, or they can be created using the **Expression Builder**
- When entering a calculation, you can zoom to expand the input area.
  Press **Shift F2** to zoom in on a text box
- You can change the properties of a calculated field by right clicking on the calculation and choosing **Properties**

Expression Rules

- When the expression refers to other fields in the QBE grid, their field names must be enclosed in square brackets: **[field name]**. You can label fields and use their labels instead
- If you want to refer to a field in another table (or if you have a field by the same name in multiple tables), then the name must be preceded by the table name, also in brackets, and separated from the field name by an exclamation point: **[table name]![field name].**
- To assign a heading to the column, precede the calculation with a label followed by a colon **label:[table name]![field name]**. If you don’t assign a label to a calculated field, Access will assign one of the forms (Expr1, Expr2, ...) for you
- The **&** operator can be used to concatenate text fields into one field.
- Standard mathematical operations can be used in expressions (+, -, *, /, ^)

Activity 6: Performing Calculations in a new Field

In this activity you will create a new column that we’ll call “**Age**.” The data in this new column will be derived from a complex calculation, which uses the **Date of Birth** field to determine a student’s current age. This calculation comes from a web page, “Working out Someone’s Age,” which can be found at this URL: http://www.fontstuff.com/access/acctut05.htm

1. Click on **Query Design** in the **Create** ribbon.
2. Add the **Student** table.

3. Close the Show Tables dialog box.
4. Add the **First Name**, **Last Name**, **Student ID**, and **Date of Birth** fields to the design grid.
5. In the field row of the 4th column enter the following expression (hint: use zoom mode – the parentheses are spread out just for better legibility) .
   Age: Int ((Date () – 9 [Date of Birth])) / 365.25)
6. Sort the calculated field **Ascending**
4. Run the query
5. Close the query and save it with the name **StudentAges.**

The examples on the web page showed two decimal places in the results. We're using the INT function here to reduce the age to a decimal value.

Grouping Records

You can group records to perform calculations over the collection of records in each group. One way to do this is through a Crosstab Query, which lets you define a two-dimensional array of groups based on pairs of values from two specified fields. Here, we will look at grouping in ordinary Select Queries, since creating such queries can be a prerequisite to designing certain reports.

The main steps are these:

1. Enable grouping by clicking on the **Totals** tool in the Show/Hide section of the Design ribbon.
2. Add to the query, in order, the fields that determine the groups you want to use to subdivide the data. Select "Group By" in the Total cell for each of those fields.
3. In the Total cell of the field you want to analyze in each group, choose the function to be applied to the collection of field data for that group. A list of available functions follows.

Functions Available

A standard set of calculation functions is available within Access to perform calculations on groups of records or to be included in expressions. The next table indicates what each function does and to what data types it can be applied.

Function	Computes, over each group	AutoNumber	Currency	Date/Time	Number	Yes/No	Text	ALL
Sum	The total of the values in the field	X	X	X	X	X		
Avg	The average of the values in the field	X	X	X	X	X		
Min	The lowest value in the field	X	X	X	X	X	X	
Max	The highest value in the field	X	X	X	X	X	X	
Count	The number of non-blank values in the field							X
StDev	The standard deviation of the values in the field	X	X	X	X	X		
Var	The variance of the values in the field	X	X	X	X	X		
First	The value from the first record							X
Last	The value from the last record							X

To use these functions:

1. Create your query as you normally would
2. Choose **Totals** from the Design ribbon
3. For the field that you want to use a function in, choose the function from the list in the **Total** row (click in the cell then click on the down arrowhead).

Note: If you have many fields in the query aside from the one that is aggregated, then you may find that the groups you generate each have just one record. As a rule, you should include in the query only the fields that determine the grouping, along with the one that is aggregated.

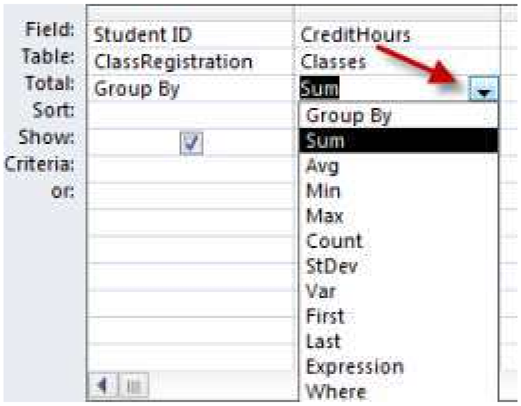
Activity 7: Grouping CreditHours into a Total

Every course in the Classes table has an associated number of credit hours, which is recorded in the field **CreditHours**. The ClassRegistration table contains records of individual students enrolled in individual classes for the current semester. In this activity we will individual student records in the ClassesRegistration table so that we can calculate the total number of credit hours the student is taking for the semester. We'll apply a minimalist approach, and just use **StudentID** and **CreditHours** in this query.

1. Click on **Query Design** in the **Create** ribbon.
2. Add the **Classes** table and the **ClassRegistration** table.
3. Close the Show Tables dialog box.
4. Add the field **Student ID** from the ClassRegistration table and **CreditHours** from the Class table to the design grid. If you ran the query now, you would get results for 2907 individual student records.

5. In the Show/Hide group of the Query Tools Design tab, click on the **Totals** command. After clicking the Totals command, the command button will be highlighted, and you will see a new “Total:” row added as the 3^{rd} row of the design grid. The text “Group By” appears beneath each Field column.

6. We want to group the CreditHours by each Student ID. Below the CreditHours field, click the right side of the “Group By” text in Total cell in order to display a menu. Select **Sum** from this menu.
7. Run the query (you should have 941 records). In the output, CreditHours is now displayed as **SumOfCreditHours**.
8. Save this query as **StudentCreditHours.** We can recall this query later for use in a Make Table Query.



Field:	Student ID	CreditHours
Table:	ClassRegistration	Classes
Total:	Group By	Sum
Sort:		
Show:	☑	
Criteria:		
or:		

Parameter query

A parameter query prompts you for the selection criteria when you run the query. To create a parameter query:

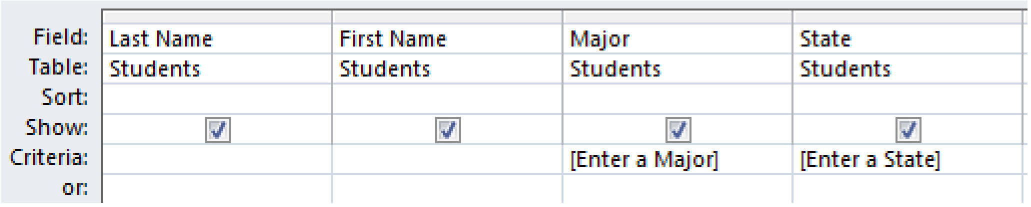
Create a normal select query but omit the criteria. In place of the criteria, place a parameter (generally in the form of a question) inside square brackets. This text will be your meaningful prompt to enter information. Repeat this for each field that you want to define as a parameter

Notes on Parameters

- If you have more than one parameter in a query, you will be prompted for the input in the order that the fields are entered into the Query Parameter dialog window
- You can use a wildcard in your parameter specification. When you enter your parameter into the criteria area, add the work **like** in front of the criteria – **Like [Country?]**

Activity 8: Parameter Query

1. If you saved your earlier **FindGeoMajors** query, open it back up in design view.
2. Under the Major column, replace the original “Geology” criteria with “[Enter a Major]” being sure to include the square brackets.
3. Under the State column, add “[Enter a State]” in the Criteria row. Again, be sure to use square brackets.



Field:	Last Name	First Name	Major	State
Table:	Students	Students	Students	Students
Sort:				
Show:	☑	☑	☑	☑
Criteria:			[Enter a Major]	[Enter a State]
or:				

4. Run the query. You will be prompted to supply a Major, followed by another prompt to type in a state abbreviation. Your output will obviously depend upon whether or not any existing records will match your choices.
5. Close the query. If you want to save it, be sure to save it as a new object, because you will otherwise overwrite your original **FindGeolMajors** query.

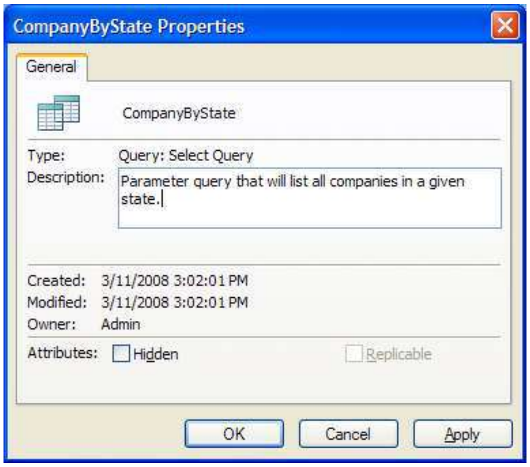
Query Properties

The Query Properties window gives you a place to provide a longer description of a query than can be achieved conveniently using the query's name.

4 Go to the list of Access Objects in the Navigation Pane that normally appears on the left side of the Access window.

4 Right click on the query and choose Object Properties from the shortcut menu.

4 Enter a meaningful description and click on OK.



CompanyByState

Type: Query: Select Query

Description: Parameter query that will list all companies in a given state.

Created: 3/11/2008 3:02:01 PM

Modified: 3/11/2008 3:02:01 PM

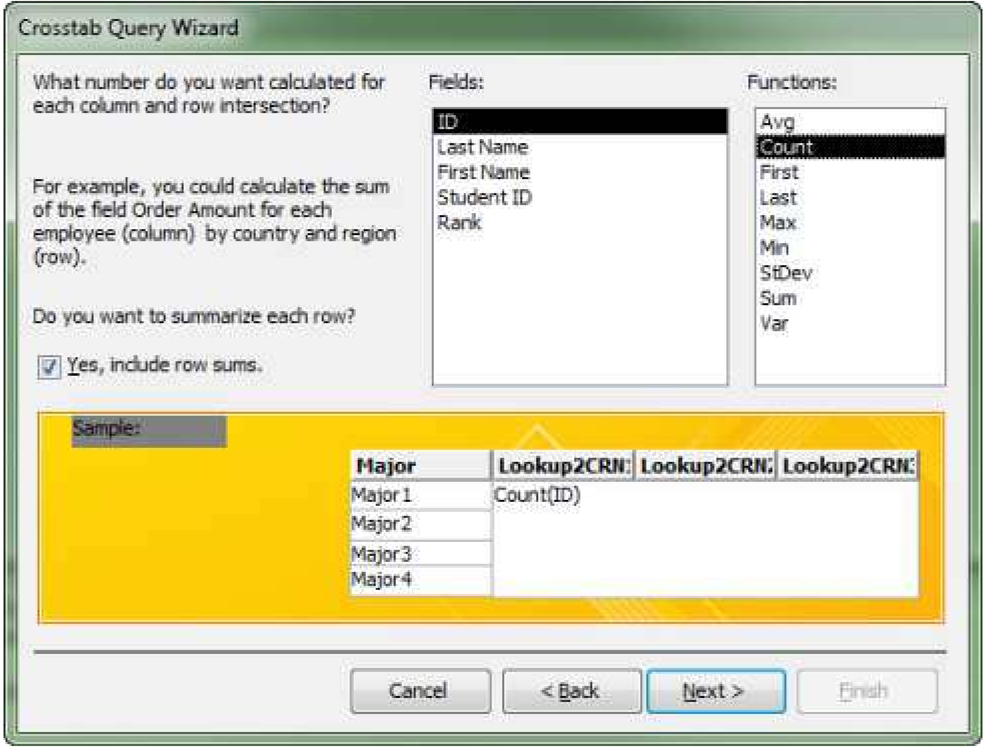
Owner: Admin

Attributes: □ Hidden □ Replicable

Crosstab Query

You can use this type of query to summarize a large table or query dynaset, with data aggregated and arrayed based on the values in chosen row and column fields. If you plan to perform this query on another query, create the base query first.

1. Click on the **Query Wizard** tool on the Create ribbon.
2. Select **Crosstab Query Wizard** in the first window to open, and click OK.
3. Pick a query or table. The ClassRegistration table, which contains nearly 3,000 records, is a good candidate. We can look at which classes are being taken by different majors.
4. Choose a row field. I chose **Major**.
5. Choose a column field. I chose **Lookup2CRN,** which will give the CRN
6. Choose a data field. I chose **ID** as the field and **Count** as the function, because I wanted to count by the number of records, which represents class registrations.



What number do you want calculated for each column and row intersection?

For example, you could calculate the sum of the field Order Amount for each employee (column) by country and region (row).

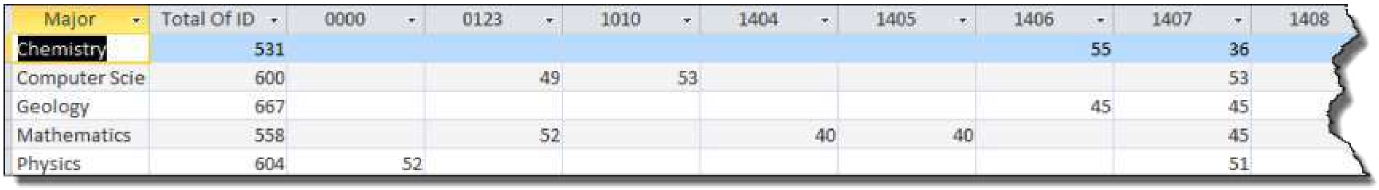
Do you want to summarize each row?

☑ Yes, include row sums.

Sample:

Major	Lookup2CRN:	Lookup2CRN:	Lookup2CRN:
Major1	Count(ID)		
Major2			
Major3			
Major4			

7. Name the query to save it and choose to view its results. As you can see, the query produces a table with a row for each major, a column for each course number (CRN), and in each cell of the table the count of enrollments. The “Total of ID” column gives the total count of enrollments by major.



Major	Total Of ID	0000	0123	1010	1404	1405	1406	1407	1408
Chemistry	531						55	36	
Computer Scie	600		49	53				53	
Geology	667						45	45	
Mathematics	558		52		40	40		45	
Physics	604	52						51	

Action Queries

Unlike the Select query, which looks for certain types of records, Action queries are used to change or delete data in a table. This category includes Update, Delete, Append, and Make Table queries.

Since these actions are very fast and could damage your files if they are not constructed correctly, you should always back up your data before using them.

Update queries add a special Update row to the QBE grid. After entering the fields that are to be updated in the grid and specifying the selection criteria, the new value is placed in the Update cell. If the value in the field matches the criteria, it is replaced with the value in the Update cell. Note that the value in the Update cell could also be the result of a calculation.

Append queries add a special *Append To* row to the QBE grid. Fields and criteria are entered into the grid to select the records from the table that are to be appended to another table. The *Append To* row is used to specify the table that the records are to be appended to. This could be another table in this database or it could be in another database.

The **Delete** query allows you to specify the fields and criteria that are to be used to select records for deletion. A special form of this query allows you to delete records from multiple tables.

The **Make-Table** query allows you to create a new table from the results of a query. The fields and criteria are used to select records for the new table. Access prompts you for the name of the new table.

A combination of these queries is frequently used to complete a change to a database. A Make-Table query might be used to create a copy of completed records from a table. A Delete query might then be used to remove those records completely from the original table. Finally, an Append query might be used to merge the copy of completed records that were deleted from the active data into a table that contains other completed records.

For all of the action queries, a typical design sequence would be the following:

1. Create a select query that determines the set of records to be acted upon.
2. Test the select query thoroughly to be sure that the right set of records is selected.
3. In Query Design View, click on the appropriate tool in the Query Type section of the Query Design ribbon to convert the select query to the new type.
4. Complete the specification of the action query, if necessary, by providing the additional information needed for the particular query type you are creating.
5. Run the action query and check the results.

Activity 9: Make an Update Query

It has been decreed that references to the “School of Arts and Sciences” be changed to “Eberly College of Arts & Sciences” or ECAS for short. A field in the Faculty table – **School**/**Program Name** – must have its values changed in order to reflect this.

1. Right-click over the **Faculty** table in the navigation pane, and select **Copy** from the shortcut menu.
2. Right-click again in the navigation pane, and select **Paste** from the shortcut menu. In the Past Table As dialog, which appears, accept the suggested name “Copy of Faculty” and make sure the Paste Option radio button is set to “Structure and Date.” Click **OK** to create the copy of the table.
3. Click on **Query Design** in the **Create** ribbon.
4. Add the **Copy of Faculty** table.
5. Close the Show Tables dialog box.
6. Add the **School**/**Program Name** field to the design grid. This Update query does not need to modify any other fields, so it is not necessary to include them.
7. In the Criteria box beneath **School**/**Program Name,** type “Arts and Sciences.” Be sure to include the quotation marks, because otherwise Access will think you are combining two criteria with an AND statement.
8. Run the query to be sure that your criterion is working properly, and return to Design View.
9. Click on the Update command button, which is in the Query Type group of the Query Tools Design ribbon. A new row, “Update To” has been added to the design grid, replacing “Sort” beneath the “Table” row.
10. In the “Update To” box, enter “ECAS” and click on the Run command in the Query Tools Design ribbon.
11. A dialog box will warn you that you are about to update 57 rows, and that this change is irreversible. Click **Yes** and continue. Nothing else will appear to happen, but if you now open the **Copy of Faculty** table, you should see that “ECAS” has replaced “Arts and Sciences” in the School/Program Name field.
12. If you save the Update query at this point, it will appear in the Navigation pane with a different icon – one that looks like a miniature Update command button. This serves to warn you that this is an Action query that can alter your data.

Saving Query Results in a Table

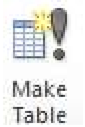
The dynaset that results from a query can be saved by creating a new table to hold the data. This is done by creating a Make-Table query. When a Make-Table query is run:

- Access checks to see if there is an existing table with the name that you have specified. If one exists, it will be deleted. If you have not changed the default settings, you will see a warning dialog window before this happens.
- Data is selected according to the criteria.
- The data is pasted into the new table. If you have not changed the default settings, you will see a warning dialog window before this happens.

Activity 10: Make Table Query

The query **StudentCreditHours**, which we had previously made in *Activity 7: Grouping CreditHours into a Total***,** can be related to the Student table and thus provide us with a calculated value of credit hours that a student is taking for a given period. In order to include this we must convert the query results into a table. Note that **StudentAges**, a query you created in *Activity 6: Performing Calculations in a new Field,* would make another fine candidate for a table.

1. Locate the query **StudentCreditHours**, which you had previously made in *Activity 7: Grouping CreditHours into a Total***.** Open it in Design View.
2. Click **Make Table** from the Query Tools Design ribbon.



3. Enter a name for the table. To avoid confusion, use a different name than any object that already exists in the database.
4. Click on **OK**
5. Run the query. Running the query is the only way that the table will be created.
6. Close the query and save it with the name **MakeTable_StudentCreditHours.**
7. Notice the different icon that this query has. It looks like the Make Table command icon. This gives you a visual cue – and a warning – that this query will perform an action.
8. Open the new table in the list of tables, and review the contents of the new table. It should appear the same as the original query output.
9. Close the table.

Note: The table you create is a static snapshot of the database. If you update or add to the original table the query was based on, this snapshot will not change.

Find Duplicates Query

You can use a Find Duplicates Query to locate records that are redundant in some way. For example, if you think that a field can be a key for a table, but data in the table is preventing you from making that field a key in Access, then it may be because a record has inadvertently been duplicated. This type of query might help you find those problematic records.

To create this type of query, proceed as in the following example:

1. Click on **Query Wizard** in the Create ribbon.
2. Click on Find Duplicates Query Wizard and then on OK.
3. Select a table and then on **Next**.
4. Select fields that might be duplicated and then click **Next**.
5. Select any additional fields that will be helpful in your investigation, and again click **Next**.
6. Provide a name for the new query, and click on **Finish**.

If the resulting dynaset is empty, then no two records have the same value in all of the fields you selected in step 4. If the dynaset is not empty, then you can review the results and decide whether to delete some of the data or alter your plans for the design of the table.