Understanding Queries

Queries are questions you ask of your database. They allow you to select certain fields out of a table, or pull together data from various related tables and display it together. Queries can be used to perform calculations, and to edit the data in your tables. Queries can also form the basis of forms and reports.

Create a Select query in Design View

1. Click the Create tab to view the Create Ribbon:

and click the Query Design button in the Queries group to create a new table in Design View.

2. The Show Table box will be displayed; add tables by double clicking them. Tables appear in the upper “tray” of the Query Design View window; existing relationships between tables are visible.

3. Double click field names in the tables to put them in the query. The fields appear in the grid below the “tray” where the tables appear.
   a. Delete a column by clicking the column selector (thin gray rectangle at the top of each column) and pressing the Delete key.
   b. Insert a column by dragging the desired field from the Field list and...
dropping in on the column where you want it to appear. Existing fields shift right to make room.
c. Move a column by dragging it by its column selector and dropping it in its new location.
d. Hide a field by unchecking its checkbox in the Show row.

4. Click the Run button ( ) to run the query and view the results. Query results are displayed in an answer table. A new answer table is generated each time the query is run.

5. Save the query by clicking the Save button and giving the query a name.

Notes on Query Results

- If a query contains fields from two or more related tables, the results will only include records that have related records in all the tables. (In other words, in the query shown above, customers without records in the Orders table will not appear in the query results.) To display such records, use an outer join (see page 6) in your query.
- WARNING! Any change you make in a query (adding, deleting or editing data) WILL BE CHANGED IN THE TABLES TOO!!!

Sort query results

In Design View, click the Sort row in the column you wish to sort by. A dropdown arrow appears; choose Ascending or Descending. If you sort by more than one column, the sorts will be done left-to-right.

Create a concatenated field in a query

Concatenated fields combine information from two or more fields into one. Ex: Combine first and last names into one field, called “FullName”. In Design View, click in a blank column and type the following:

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

(Type a space between the two sets of quotation marks, and use square brackets around the field names.)
Create a calculated field in a query

In Design View, click in a blank column and type a mathematical expression, typing field names in square brackets. Any of the normal mathematical operators (+ - * / ^) can be used. If desired, give the expression a name as in the sample below. The following expression calculates a total cost by multiplying quantity times price:

Item Total: [Quantity] * [Price]

Use the Expression Builder to create calculated or concatenated fields

You can simplify the process of creating concatenated fields and other expressions with the Expression Builder. Create a blank column in your query grid and then click the Builder button in the Query Setup group on the Query Design Ribbon. The Expression Builder opens.

Type the expression in the large box at the top. One advantage is that you have way more room to type and see your expression than you have at the top of a query grid column!

A second advantage is that you can add fields to the expression by double clicking them in the Expression Categories column rather than by typing them. This removes the possibility of typos.

Summarize data with a Totals query

- In Query Design View, click the Totals button (∑) in the Show/Hide group.
- A new line, called Total, will appear in the query. All the fields will be designated “Group By”. Click the drop down arrow in the desired field, and choose the calculation you want (sum, average, etc.)
- Run the query.

All query fields that are not being summarized must match for the records to total properly! (see illustration, next page.)
Specify criteria in a query (Filtering)

In Query Design View, click in the criteria row for the field you want to filter, and type the criteria (ex: type “OH” in the criteria row of the State field). When you run the query, only records matching the criteria will be displayed.

This Totals query does not actually total the orders, because the different values in the ProductID, Yarn Line, Price and Quantity fields prevent Access from “collapsing” or grouping the fields for the order.

With those fields removed, the query now totals each order without difficulty.
Using Multiple Criteria

You can type criteria in more than one field.

- Criteria typed on the same line must all be met for a record to be displayed (“AND” logic). Ex: “OH” in the state field and “Smith” in the LastName field pulls up only Ohio-dwelling Smiths.
- If criteria are typed on different lines, records matching any of them will be retrieved (“OR” logic). Ex: “WA” and “OH” on different lines in the State field retrieves records for both states.

Criteria Ranges

Criteria ranges broaden and customize a search. Type a criteria range instead of a single value in the Criteria row of the query grid.

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Example</th>
<th>Description and Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;</td>
<td>&lt;14</td>
<td>Finds records where the field is less than a certain value. Records will have values under 14.</td>
</tr>
<tr>
<td>&lt;=</td>
<td>&lt;= #2/2/2006#</td>
<td>Finds records where the field is less than or equal to a certain value. Records will have dates on or before February 2, 2006 Note: dates are enclosed between # signs.</td>
</tr>
<tr>
<td>&gt;</td>
<td>&gt;100</td>
<td>Similar to above: Finds records where the value is greater than 100. The second expression displays records where the value is greater than or equal to 100.</td>
</tr>
<tr>
<td></td>
<td>&gt;=100</td>
<td></td>
</tr>
<tr>
<td>=</td>
<td>=“Diana”</td>
<td>Finds records where the value in the field is Diana. Note: text is enclosed in double quotes.</td>
</tr>
</tbody>
</table>

Expressions like these may be combined - for example, a criteria of <49 Or >=100 would return records with values below 49, or 100 or above, but not ones with values between 49 and 99.99.

<table>
<thead>
<tr>
<th>Between __ And __</th>
<th>Between 1 and 4</th>
<th>Finds records where the field is between the endpoints.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is Null</td>
<td>Is Null</td>
<td>Finds records where the field is empty.</td>
</tr>
<tr>
<td>Is Not Null</td>
<td>Is Not Null</td>
<td>Finds records where the field is not empty.</td>
</tr>
<tr>
<td>Like</td>
<td>Like U*</td>
<td>Finds records where the criteria match parts of the field.</td>
</tr>
<tr>
<td></td>
<td>Like &quot;[U-X]*&quot;</td>
<td>The * wildcard stands for any number of characters, and</td>
</tr>
<tr>
<td></td>
<td>Like &quot;<em>Korea</em>&quot;</td>
<td>the ? wildcard stands for a single character.</td>
</tr>
<tr>
<td></td>
<td>Like &quot;*ina&quot;</td>
<td>The sample criteria will return records where the field</td>
</tr>
<tr>
<td></td>
<td>Like &quot;Chi??&quot;</td>
<td>1) starts with U; 2) starts with U, V, W or X, 3) contains</td>
</tr>
<tr>
<td></td>
<td>Like &quot;*4.99&quot;</td>
<td>the letters ‘Korea’; 4) ends in the letters ‘ina’ as in ballerina; 5) has five letters and begins with the letters ‘Chi’ as in Chile or China; 6) is a number and ends in the digits ‘4.99’ as in 24.99.</td>
</tr>
</tbody>
</table>

Create a parameter query

Parameters make queries flexible by letting users choose their own criteria each time they run the query. In the criteria box, type a phrase in square brackets; Access will use this phrase to prompt the user for a value. Ex: In a State field, type [Enter a state]. If the user types CA, only California records will be retrieved.
Specify a range in a parameter query

In the criteria box, type the following:

Between [___] And [___]

Inside the brackets, type text telling the user what kind of information to enter.

Inner Joins vs. Outer Joins

Excel normally uses *inner joins* in its queries. These queries return only those rows from both tables in the join that match on the joining field (e.g., customers in the Customers table with corresponding orders in the Orders table.)

If you need to display *all* customers, with or without orders, as well as any available order information, you need to use an outer join.

In the query tray, double click the join line between two related tables. The Join Properties dialog box will appear:

Select button 2 or 3 depending on which field’s records you want to display completely.

Action Queries

Action queries modify groups of records in one fell swoop. Types of action queries include:

- Update Queries (modify all records in a table that meet specific criteria – ex: change the tax rate for all Ohio customers from 5.75% to 6%)
- Delete Queries (remove records from a table – ex: delete all customers whose last order date is more than 2 years ago)
- Append Queries (add records to an existing table – ex: at the end of the year, add the year's orders to the end of an archive table)
- Make-table Queries (create a new table – ex: extract a subset of records into a new table, for a subordinate who needs Ohio customers only)

Note: Undo will NOT undo the results of action queries!
Create an update query

1. Create a new table in Design View. Add the table you will be updating.
2. Add the following fields to the query:
   a. The field you will be updating and
   b. The fields you need as criteria.
      For example, if you are changing the sales tax for all businesses in Cuyahoga County, you would need to add a Sales Tax field and a County field to your query.
3. Change the query type to an update query by clicking the Update button in the Query Type group on the Query Tools Design toolbar:

   ![Query Tools Design Toolbar](image)

4. In the Criteria row of the query, type criteria to select the records that will be changed.
5. In the Update To row of the query, add an expression to update the desired fields. (example: `[Price]*1.1` will raise all the values in the Price field by 10%.)
   (Note: You can use parameters in the Criteria and/or Update To rows.)
6. Click the View button ( ![View Button](image) ) to preview records that will be affected; click it again ( ![View Button](image) ) to return to design view.
7. Click Run to run the query. Access will show a warning box telling you how many records it will modify; click Yes to continue updating the records.

![Update Query Result](image)
Other Specialized Queries

Instructions for creating and running these special query types can be found in the Access help manual by clicking the Help button in the upper right corner of the screen (Internet connection required).

Find Duplicates query: Lets you locate duplicate records in a table by displaying a field containing duplicate values alongside other fields for comparison, to help you tell if the record is really a duplicate.

Find Unmatched query: Lets you compare two tables and identify records in one of the tables that have no corresponding records in the other table.

Crosstab query: This is a special type of Totals query that presents summary information in a compact format that is similar to a spreadsheet.

Union query: Lets you review all of the data that is returned by several similar select queries together, as a combined set.

Practice Assignment:

1. Open your “NEOhio Outreach” database. Create a query that pulls up each donor’s contributions:
   a. Create a new query in Design View (click the Create tab and click the Query Design button in the Other group).
   b. Select and add the following tables: Contributions and Donors; close the Show Table box.
   c. Add the following fields into the query: all fields from the Contributions table (use the * to add all fields at once!), and the Last Name, First Name, Address, City, State and Zip fields from the Donors table. (The only fields you are leaving out are the Donor ID and Phone Number fields from the Donors table.)
   d. Sort the query into ascending order by last name, then within each last name, sort into ascending order by first name.
   e. Save the query as Donor Statements. Run the query to be sure it works, then close the query.

2. Create a query that selects a list of donors and calculates the total contribution and average contribution amounts for each donor. Set parameters that enable the user to choose a date range.
   a. From within your “NEOhio Outreach” database, create a new query in Design View. Select and add the following tables: Contributions and Donors; close the Show Table box.
   b. Add the Last Name field, and set the query to sort by this field, but not to display it. In the next column, create a Full Name field that will show both the first name and the last name. Click Run to check that only the Full Name field is displayed. Return to Design View.
   c. Place the Amount field in each of the next two columns, then click the Totals button. In the first Amount field, calculate the Sum; in the second one, calculate the Average.
   d. Place the Date field in the next column, and choose Where in the Total row. In the criteria row, type an expression that will select dates between a user-entered starting date and ending date.
   e. Save your query as Contribution Report.