

Microsoft® Access 2003

II: Relationships & Queries

A Workshop for San Diego State University Faculty and Staff



Where to Find Help When You Need It

Help from your Division/College's Computer Consultant

Some divisions and colleges have computer consultants assigned to them. You can contact these consultants when you need help. To determine if you have a consultant assigned to your division or college, look to: <http://rohan.sdsu.edu/~facstaff>

Help from the BATS Web Page

BATS (Baseline Access, Training and Support) is a California State University initiative to provide all students, faculty, and staff with "baseline" access to information resources via networks, training in the uses of baseline hardware and software systems, and ongoing professional and technical support for utilization of computer resources at San Diego State University. You can access the BATS Web Page by pointing your browser to: <http://rohan.sdsu.edu/~bats/>

Help in the San Diego State University, Faculty Room

The Faculty Room is staffed Monday through Friday with computing consultants who will try to answer your questions.

Location: Adams Humanities, 1109
Phone Number: x45727
Semester Hours: 7:30am – 6:00pm Monday -Thursday
7:30am – 4:30pm Friday
Semester Intersession: 7:30am – 4:30pm Monday – Friday

Help from the Faculty Computing Help Line

Phone Number: x41348 **E-mail:** helpline@mail.sdsu.edu
Semester Hours: 7:30am – 6:00pm Monday – Thursday
7:30am – 4:30pm Friday
Semester Intersession: 7:30am – 4:30pm Monday – Friday

Help from the Staff Computing Help Line

Phone Number: x40824 **E-mail:** staffhelp@sdsu.edu
Semester Hours: 7:30am – 6:00pm Monday – Thursday
7:30am – 4:30pm Friday
Semester Intersession: 7:30am – 4:30pm Monday – Friday

TABLE OF CONTENTS

RELATIONSHIPS	1
Types of Relationships	1
Creating A Relationship	1
Referential Integrity.....	2
QUERIES	4
Creating A Query	6
Entering Criteria	7
Using Compound Criteria.....	7
Sorting Data in a Query	8
Using Computed Fields in a Query	8
Calculating Statistics	8
Grouping	8
Saving a Query.....	9

RELATIONSHIPS

Once you've set up different tables for each subject in your database, you need a way of telling Microsoft Access how to bring that information back together again. The first step in this process is to define relationships between your tables. After you've done that, you can create queries, forms, and reports to display information from several tables at once.

Types of Relationships

The type of relationship Microsoft Access creates depends on how the related fields are defined:

- A **one-to-many** relationship is created if only one of the related fields is a primary key or has a unique index. This is the most common relationship.
- A **one-to-one** relationship is created if both of the related fields are primary keys or have unique indexes. This relationship doesn't occur very often.
- A **many-to-many** relationship is really two one-to-many relationships with a third table whose primary key consists of two fields - the foreign keys from the two other tables. This relationship is the most complex and is used very infrequently.

Creating A Relationship

To define Relationships between tables:

1. Close all tables. You can't create or modify relationships between open tables.
2. Display the Database window.
3. Click **Relationships** on the toolbar.
4. If your database doesn't have any relationships defined, the **Add Tables/Queries** box will automatically be displayed. If you need to add the tables you want to relate and the Add Table dialog box isn't displayed, click **Show Table** on the toolbar. If the tables you want to relate are already displayed, skip to step 6.
5. Double-click the names of the tables you want to relate, and then close the Add Tables/Queries dialog box.

Microsoft Access II: Relationships and Queries

6. Drag the field that you want to relate from one table to the related field in the other table.

Note→

In most cases, you drag the primary key field (which is displayed in bold text) from one table to a similar field (often with the same name) called the foreign key in the other table. The related fields don't have to have the same names, but they must have the same data type (with two exceptions) and contain the same kind of information. In addition, when the matching fields are Number fields, they must have the same Field Size property setting.

7. The **Relationships** dialog box is displayed. Check the field names displayed in the two columns to ensure they are correct. You can change them if necessary.
8. Click the **Create** button to create the relationship.
9. When you close the **Relationships** window, Microsoft Access asks if you want to save the layout. Whether you save the layout or not, the relationships you create are saved in the database.

Referential Integrity

Referential integrity is a system of rules that Access uses to ensure that relationships between records in related tables are valid, and that you don't accidentally delete or change related data. You can set referential integrity when all of the following conditions are met:

- The matching field from the primary table is a primary key or has a unique index.
 - A unique index is an index defined by setting a field's Indexed property to Yes (No Duplicates). A unique index will not allow duplicate entries in the indexed field. Setting a field as the primary key automatically designates the field as a unique index.
- The related fields have the same data type.
 - There are two exceptions. An AutoNumber field can be related to a Number field with a FieldSize property setting of Long Integer, and an AutoNumber field with a FieldSize property setting of Replication ID can be related to a Number field with a FieldSize property setting of

Microsoft Access II: Relationships and Queries

Replication ID.

- Both tables belong to the same Microsoft Access database.
- If the tables are linked tables, they must be tables in Microsoft Access format, and you must open the database in which they are stored to set referential integrity. Referential integrity can't be enforced for linked tables from databases in other formats.

When referential integrity is enforced, you must observe the following rules:

- You can't enter a value in the foreign key field of the related table that doesn't exist in the primary key of the primary table. However, you can enter a Null value in the foreign key, specifying that the records are unrelated.
 - For example, you can't have an order that is assigned to a customer that doesn't exist, but you can have an order that is assigned to no one by entering a Null value in the CustomerID field.
- You can't delete a record from a primary table if matching records exist in a related table.
 - For example, you can't delete an employee record from the Employees table if there are orders assigned to the employee in the Orders table.
- You can't change a primary key value in the primary table, if that record has related records.
 - For example, you can't change an employee's ID in the Employees table if there are orders assigned to that employee in the Orders table.

If you want to enforce these rules for a relationship, select the **Enforce Referential Integrity** check box when you create the relationship. If referential integrity is enforced and you break one of the rules with related tables, Access displays a message and doesn't allow the change.

Microsoft Access II: Relationships and Queries

You can override the restrictions against deleting or changing related records and still preserve referential integrity by setting the **Cascade Update** Related Fields and **Cascade Delete** Related Records check boxes.

- When the **Cascade Update Related** Fields check box is set, changing a primary key value in the primary table automatically updates the matching value in all related records. (In most databases this option is selected.)
- When the **Cascade Delete Related** Records check box is set, deleting a record in the primary table deletes any related records in the related table. (This option is sometimes selected, but only in certain specific cases.)

QUERIES

Access offers the ability to answer questions. The answer to these questions, and many more are found in the database, and Access can find the answers quickly. When you pose a question to Access, the question is called a query. A query is simply a question represented in a way that Access can understand.

You use queries to view, change, and analyze data in different ways. You can also use them as the source of records for forms and reports. The most common type of query is a select query. A select query retrieves data from one or more tables using criteria you specify, and then displays it in the order you want.

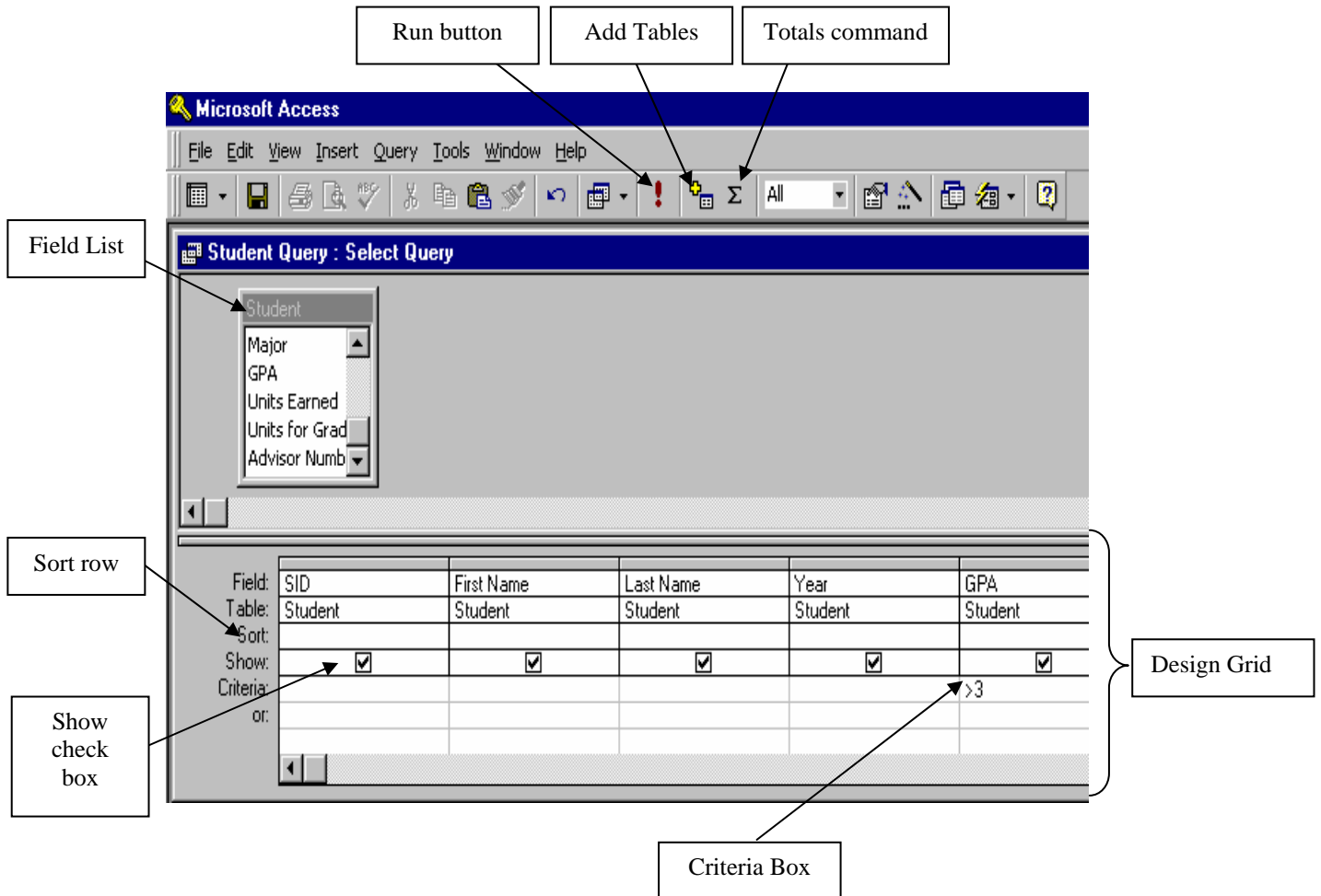
→CAUTION

Making changes to a query record or deleting a record in a query will affect the same record in the related table.

Microsoft Access II: Relationships and Queries

Design View

The design grid is used to setup a query in Design view.



There are two sections in design view:

- The top section is where the table or query will be displayed from which you are basing your new query. Inside each table or query window will be listed the field names.
- The bottom section is where the field names are moved to set up the criteria or other options to base the query on.

In the bottom section, each row will consist of a row heading to describe its purpose. You will find the following:

- **Field** – The field name is displayed here. Calculations can also be made in this cell to display the result based on an expression.
- **Table** – The table or query from which the field was selected displays here.

Microsoft Access II: Relationships and Queries

- **Sort** – You indicate sorting preferences here.
- **Show** - Check the box to display the data in the field.
- **Criteria** – Specific selection criteria is listed here.
- **Totals** - By going into the View menu and clicking **Totals**, the Totals row heading will be added to the design grid. Click the drop-down arrow to display the results of a calculation in a field, based on a predefined calculation such as, sum, average, count, minimum, maximum, standard deviation, or variance. You choose one totals calculation for each field you want to calculate.

Creating A Query

To create a query in Design View:

1. In the Database window, click the **Queries** tab, and then click **New**.
2. In the New Query dialog box, click **Design View**, and then click **OK**.
3. In the Show Table dialog box, click the tab that lists the objects whose data you want to work with.
4. Double-click the name of each object you want to add to the query, and then click **Close**.
5. If you have multiple tables or queries in the query, make sure they are connected to each other with a join line so that Microsoft Access knows how the information is related. If they aren't connected, create the join line yourself.
6. Add fields to the query by double-clicking the field names from the field list.
7. Refine your query by entering criteria, adding a sort order, creating calculated fields, computing the sum, average, count, or another type of total on the data it retrieves, or otherwise modifying the query's design.
8. To save the query, click **Save** on the toolbar. Enter a name and then click **OK**. To see the results of the query, i.e. run the query, click the **View** button on the toolbar.



Microsoft Access II: Relationships and Queries

Entering Criteria

To enter criteria, enter them on the Criteria row in the design grid below the field name to which the criterion applies.

Text

To use text data in criteria, simply type the text in the Criteria row below the corresponding field name. For example, to indicate that the student's Last Name is 'Jones', you would type Jones in the Criteria row below the Last Name field.

Wildcards

Two special wildcards are available. The first of the two wildcards, the **asterisk** (*), represents any collection of characters. For example in the Last Name field criteria, typing Sm*, means any last name that starts with Sm followed by any collection of characters. The other wildcard symbol is the **question mark** (?) which represent any single character. For example, T?m, could be Tim or Tom.

Numeric Data

To enter a number in a criterion, type the number without any dollar signs or commas.

Using Compound Criteria

Comparison Operators

If you want a result other than an exact match, you must enter the appropriate comparison operator. The comparison operators are > (greater than), < (less than), >= (greater than or equal to), <= (less than or equal to), and **NOT** (not equal to).

For a list of examples of criteria expressions see Appendix A.

To query with more than one criterion you must use a compound criteria.

Using the AND criteria

In an **AND** criterion, each individual criteria must be true in order for the compound criteria to be true. For example you would use an AND criterion to find students that are Freshman AND who are Economics majors. To combine criteria with AND, place the criteria on the same line.

Microsoft Access II: Relationships and Queries

Using the OR criteria

An **OR** criterion is true if either individual criteria is true. For example you would use an OR criterion to find students that are Freshman OR who are Economics majors. To combine criteria with OR, the criteria must go on separate lines in the Criteria area.

To order records in the answer to a query in a particular way, you sort the records. The field or fields on which the records are sorted is called the sort key.

Sorting Data in a Query

To sort the results of the query, you must specify the sort order in the Sort line of the design grid below the field that is to be sorted. If you specify more than one field to sort, the field sorted on the left will be sorted first and the one on the right will be sorted next.

Using Computed Fields in a Query

To include a computed field in queries, you enter a name for the computed field, a colon, and then the expression in one of the columns in the Field row. For example to calculate Total Price, you would type Total Price:[Price]*[Quantity]. You can type this directly in the Field row.

You most likely will not be able to see the entire entry, because the Field row is not large enough. To avoid this, select the column in the Field row, right-click to display the shortcut menu, and then click Zoom. The zoom dialog box displays where you can type the expression.

Calculating Statistics

The computations you can do are addition (+), subtraction (-), multiplication (*), or division (/). You can use parentheses in your computation to indicate which computation should be done first.



There are nine built in aggregate functions for Access. For a list of the built in aggregate functions and how they are used see Appendix B. To use any of these in a query, you include it in the Total row in the design grid. By default the Total row does not appear in the grid. To include it, click the **Totals** button on the toolbar.

Grouping

Often statistics are used in combination with grouping. For example you may want the average GPA for all students in each major of study. Grouping means creating groups of records that share some common characteristic.

In this example, when grouped by major, you will have one group of Business Administration students, one group of Finance students, and one group of Economic students. The calculation of

average GPA would be made for each group.

To indicate grouping in the query, select **Group By** as the entry in the Total row for the field to be used for grouping. Then select the aggregate function, such as average (avg).

Saving a Query

In many cases, you will construct a query you will want to use again. By saving the query you will eliminate the need to repeat the design of that query.

1. Design the query you want.
2. Click the **save** button. Type the name you want to save this query as and then point to the **OK** button.

Once you have saved a query, you can use it any time in the future by opening it. To open a saved query, click the Queries tab in the Database window, select the query, and then click Open.

The query is run against the current database, thus if changes have been made to the data since the last time you ran it, the results of the query may be different.

Appendix A Examples of Criteria Expressions in Queries

Field	Expression	Description
ShipCity	"London"	Displays orders shipped to London.
ShipCity	"London" Or "Hedge End"	Uses the Or operator to display orders shipped to London or Hedge End.
ShippedDate	Between #1/5/95# And #1/10/95#	Uses the Between...And operator to display orders shipped no earlier than 5-Jan-95 and no later than 10-Jan-95.
ShippedDate	#2/2/95#	Displays orders shipped on 2-Feb-95.
ShipCountry	In("Canada", "UK")	Uses the In operator to display orders shipped to Canada or the UK.
ShipCountry	Not "USA"	Uses the Not operator to display orders shipped to countries other than the USA.
ShipName	Like "S*"	Orders shipped to customers whose name starts with the letter S.
CompanyName	>="N"	Displays orders shipped to companies whose name starts with the letters N through Z.
OrderID	Right([OrderID], 2)="99"	Uses the Right function to display orders with OrderID values ending in 99.
CompanyName	Len([CompanyName]) >Val(30)	Uses the Len and Val functions to display orders for companies whose name is more than 30 characters long.
ShipRegion	Is Null	Displays orders for customers whose ShipRegion field is Null (blank).
ShipRegion	Is Not Null	Displays orders for customers whose ShipRegion field contains a value.
Fax	""	Displays orders for customers who don't have a fax machine, indicated by a zero-length string value in the Fax field instead of a Null (blank) value.
ShipName	Like "S*"	Orders shipped to customers whose names start with the letter S.
ShipName	Like "*Imports"	Orders shipped to customers whose names end with the word "Imports".
ShipName	Like "[A-D]*"	Orders shipped to customers whose names start with A through D.
ShipName	Like "*ar*"	Orders shipped to customers whose names include the letter sequence "ar".
ShipName	Like "Maison Dewe?"	Orders shipped to the customer with "Maison" as the first part of its name and a 5-letter second name in which the first 4 letters are "Dewe" and the last letter is unknown.

Appendix B

About aggregate functions and other options in the query design grid's Total row

Select	To find the	Use with these field data types
Sum	Total of the values in a field.	Number, Date/Time, Currency, and AutoNumber
Avg	Average of the values in a field.	Number, Date/Time, Currency, and AutoNumber
Min	Lowest value in a field.	Text, Number, Date/Time, Currency, and AutoNumber
Max	Highest value in a field.	Text, Number, Date/Time, Currency, and AutoNumber
Count	Number of values in a field, not counting Null (blank) values.	Text, Memo, Number, Date/Time, Currency, AutoNumber, Yes/No, and OLE Object
StDev	Standard deviation of the values in a field.	Number, Date/Time, Currency, and AutoNumber
Var	Variance of the values in a field.	Number, Date/Time, Currency, and AutoNumber

Select	To
Group By	Define the groups you want to perform the calculations for. For example, to show total sales by category, select Group By for the CategoryName field.
Expression	Create a calculated field that includes an aggregate function in its expression. Usually, you create a calculated field when you want to use multiple functions in an expression.
Where	Specify criteria for a field you aren't using to define groupings. If you select this option for a field, Microsoft Access will hide the field in the query results by clearing the Show check box.
First	The First and Last functions simply return the value of a specified field in the first or last record, respectively, of the result set returned by a query.
Last	Because records are usually returned in no particular order (unless the query includes an ORDER BY clause), the records returned by these functions will be arbitrary.