

Using Access 2000

A *database* is a collection of data organized in such a way to provide easy access, retrieval, and use of that data. A *database management system (DBMS)* is computer software that allows you to create a database, add information to a database, modify and delete information, sort and retrieve information, and to create reports using the information in the database. A database can be fairly simple, consisting merely of a list of names, addresses, and phone numbers, or extremely complex consisting of many tables of information. *Access 2000* is a database management system that considers a database to be a collection of one or more tables.

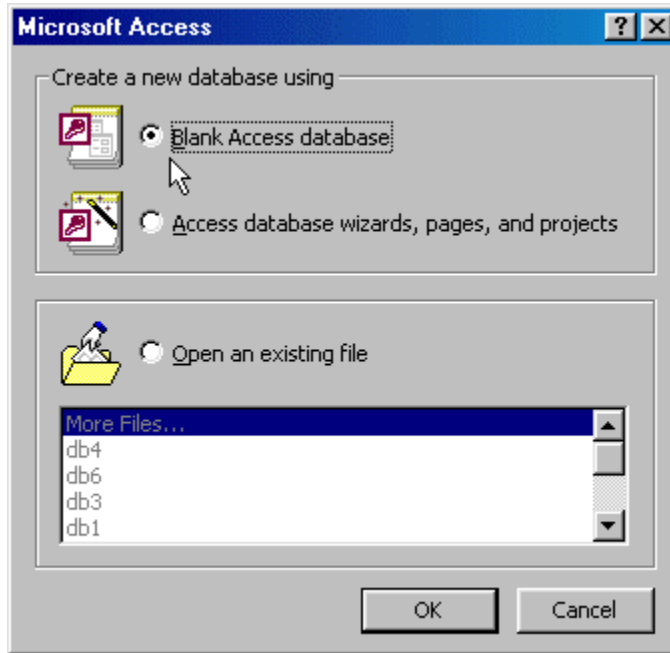
Typically, a database contains **records** of information. A record may, for example, consist of first name, last name, street address, city, state, and zip code. Each record contains **fields**. Fields are the specific parts of a record--the first name field, the last name field, the street address field, etc. In *Access*, such records are stored in **tables**. A table is a collection of information organized in rows, with each row of information corresponding to a record, and each column in the table a field. For example, consider the table below. Each row in this table represents a student record. The columns in the table represent the fields of ID, FIRST NAME, LAST NAME, and GPA, respectively:

STUDENT ID	FIRST NAME	LAST NAME	GPA
12345	Samuel	Adams	3.45
98765	Teddy	Bare	2.98
33224	Barbie	Dahl	3.23
44332	John	Doe	2.99
55555	Ella	Vader	3.50

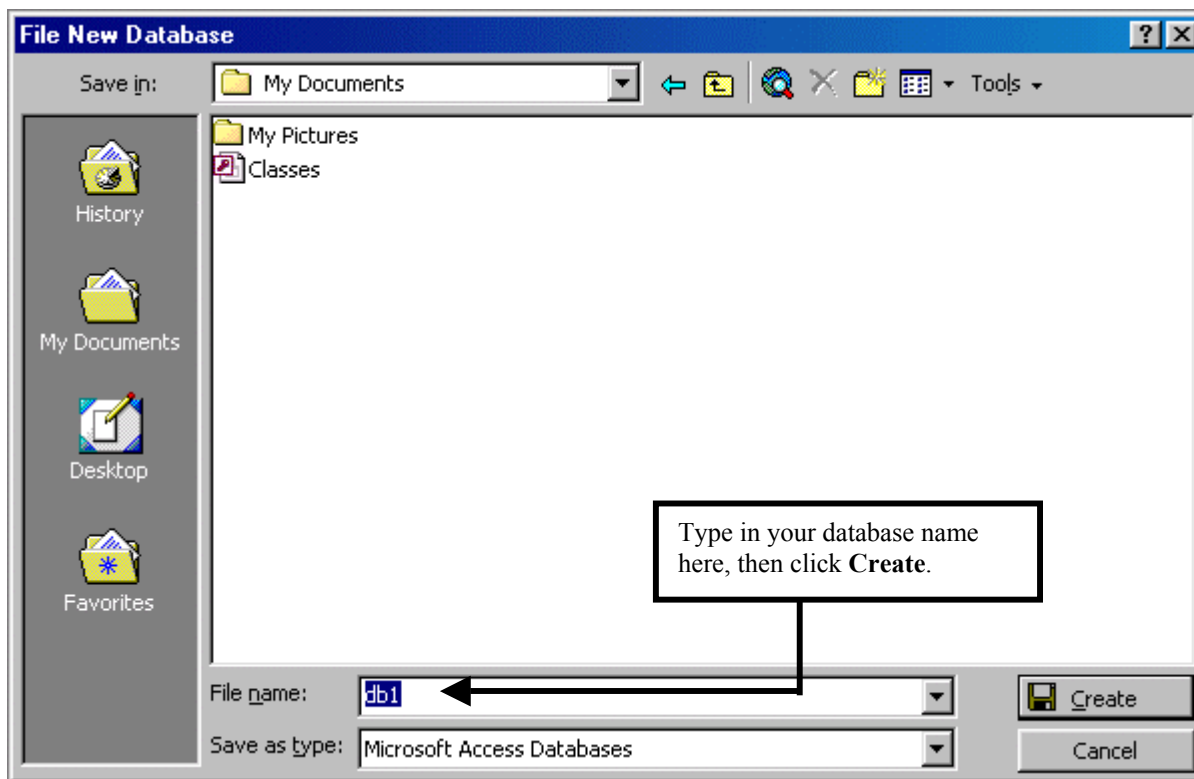
This introduction to *Access* will go over the creation of relatively simple tables, but does not go into detail about the creation of complex databases with multiple tables. Multiple tables that have a relationship between fields in the various tables (such as a database for a business containing a table of the names of its customers and their sales representative and another table containing the names sales representatives) is called a *relational database*, and is one of the strengths of *Access*. This handout is meant as a basic introduction to the program; you should consult a book on *Access 2000* for more detailed information.

CREATING A NEW DATABASE

Open *Access* by clicking on the **Start** button on the task bar in Windows 95/98, then click **Programs**, and finally on **Access 2000**. The first screen you see will give you the choice of opening an existing database or creating a blank one. Since you want to start a new database, click on the choice **Blank Database**, as shown on the screen shot below:



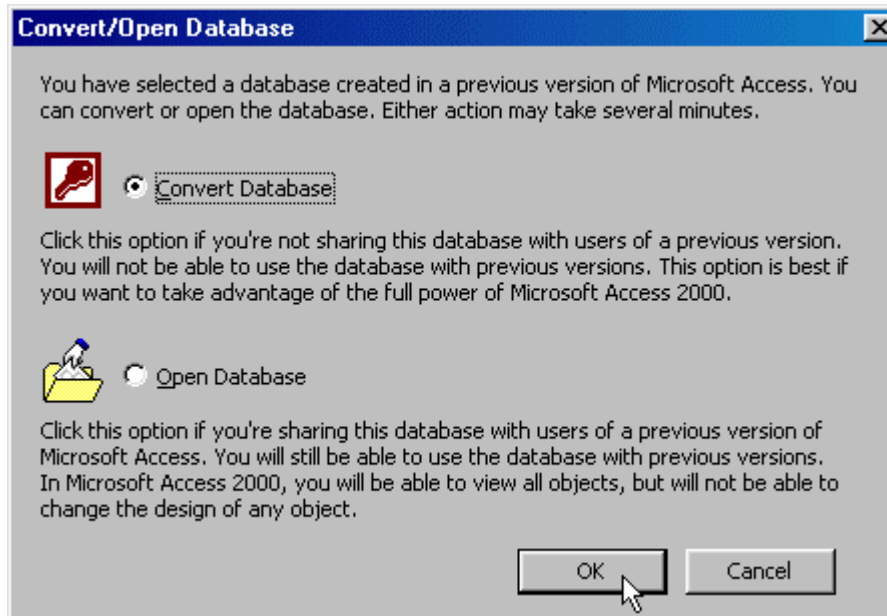
After clicking **OK**, you will see that following screen:



Give your database an appropriate name in the box, and then click the **Create** button.

NOTE FOR THOSE OPENING AN ACCESS97 DATABASE

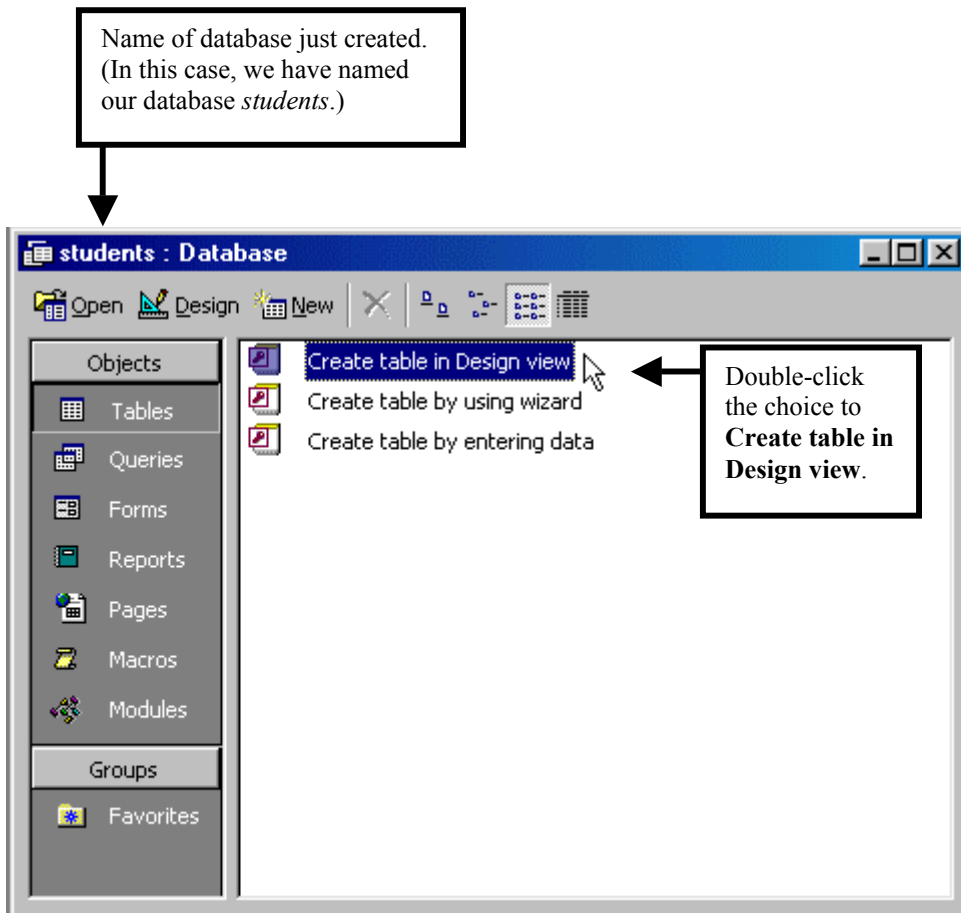
Unlike other *Office 2000* applications, if you try opening a database created with a previous version of *Access*, you will be notified immediately with the following dialog box:



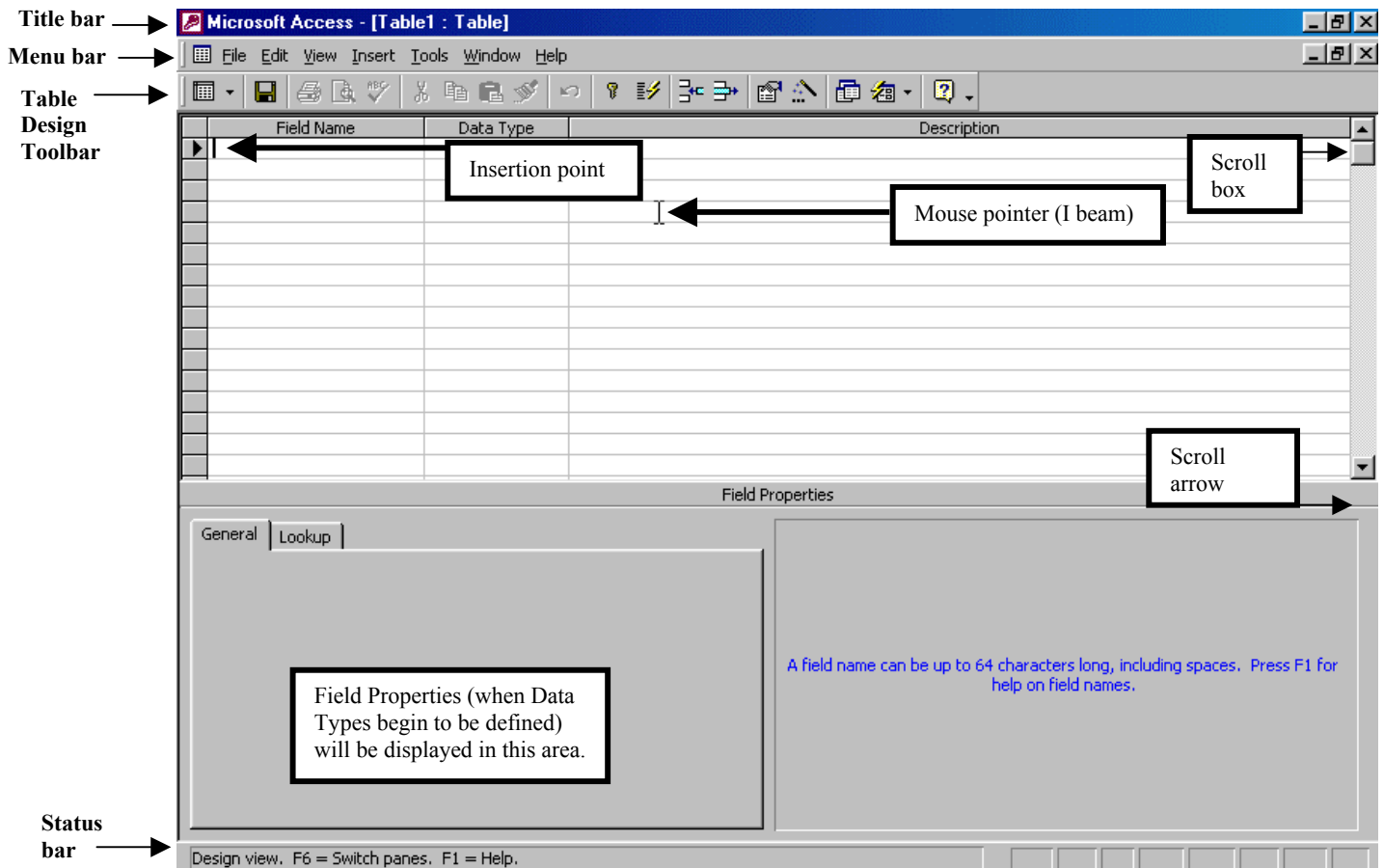
If you plan on altering the database in any way, you *must* click the first button (**Convert Database**) and have *Access 2000* convert the database to a format it will recognize. You will then be prompted to enter a *new* name for this database before it will open. (In other words, *Access 2000* will not immediately overwrite your old database with this new one; after you are satisfied that the conversion worked, you will be able to go back later and delete the old database.) You will not likely have any trouble whatsoever with this conversion process.

CREATING A NEW TABLE

After naming your database, you will get the screen below. Since a database in *Access* is a collection of one or more tables, your first task will be to create a table. For each table, you will have to define the table structure. It is only after defining this structure that you will be able to actually begin adding data. So, *Access* prompts you for creating a table first:



It is best to go with the default of **Create table in Design View**, since this gives you the greatest flexibility in designing your new table. Having some knowledge of table structure is necessary even in the simplest database, so although the next two choices may offer solutions that sound easier, in our opinion it will be far easier for you--in the long run--to design a data table from scratch. After double clicking on that choice, you will see the screen depicted on the next page.



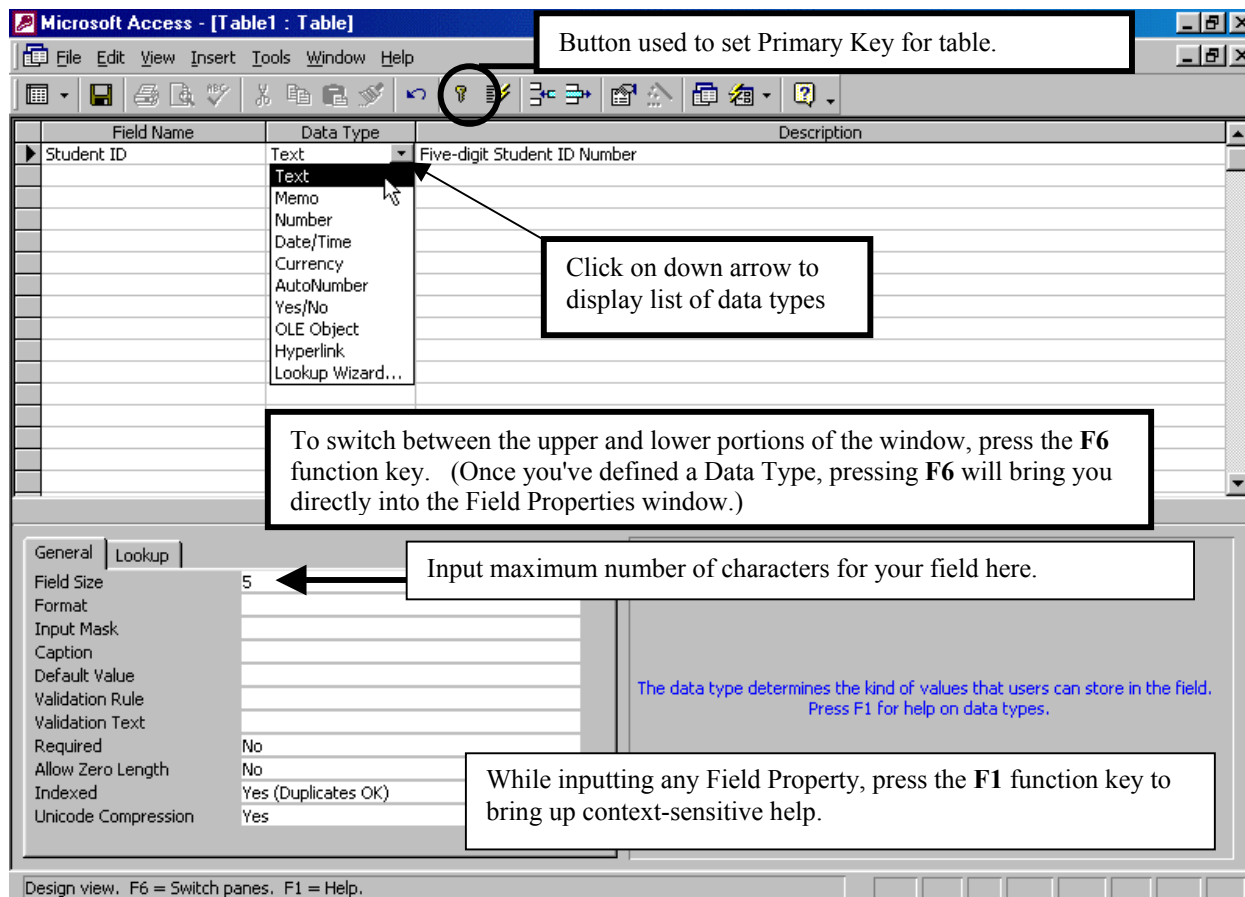
SPECIFYING A TABLE STRUCTURE

You will now begin to define the fields in the table. In the table design view, as shown above, each specific field you will be creating is in a row of the table. Each row contains three elements, with the first two being the most important: field name, data type, and description. Check the screen shot on the next page for what the table design view looks like while doing this.

- With the cursor (insertion point) in the top row in the **Field Name** column, type the name of your first field in your table.
- Press the **Tab** key on your keyboard to move the cursor over to the **Data Type** column, or click in the column yourself with the mouse.
- Click the down arrow in the **Data Type** column to select a data type. There are nine data types to choose from (these are discussed later). Click the one you wish to select.
- Note that as soon as you select a data type, the **Field Properties** box at the bottom of the screen is available for input. Make sure you are in the **General** section of Field Properties (the default). The one element you may want to take advantage of here is **Field Size**. Type in the maximum length of this field. (Be careful not to make it too short!)
- Press **Tab** again and type in an optional description, which will appear in the Status Bar portion of your screen when you begin to input data.

- Press **Tab** again or click the left mouse button in the first column (Field Name) of the next field to be defined. Continue entering each field this way until finished with your table design.

Here is what the screen looks like while you are inputting your field definitions:




Data Types

The data type you select for the field is up to you, and depends how you plan on using the field for other *Access* features such as reports or queries. For example, notice that the ID number we have in our database so far was made a *text* field, even though it contains all numbers. Why? We do not plan to manipulate that data in any way. On the other hand, our GPA field will be made a *number* field, since it is possible that for a query, we may want to limit a search to those students with GPAs of 3.25 or higher--or perhaps to even find an average of all GPAs.

The following page lists the data types available within *Access*.

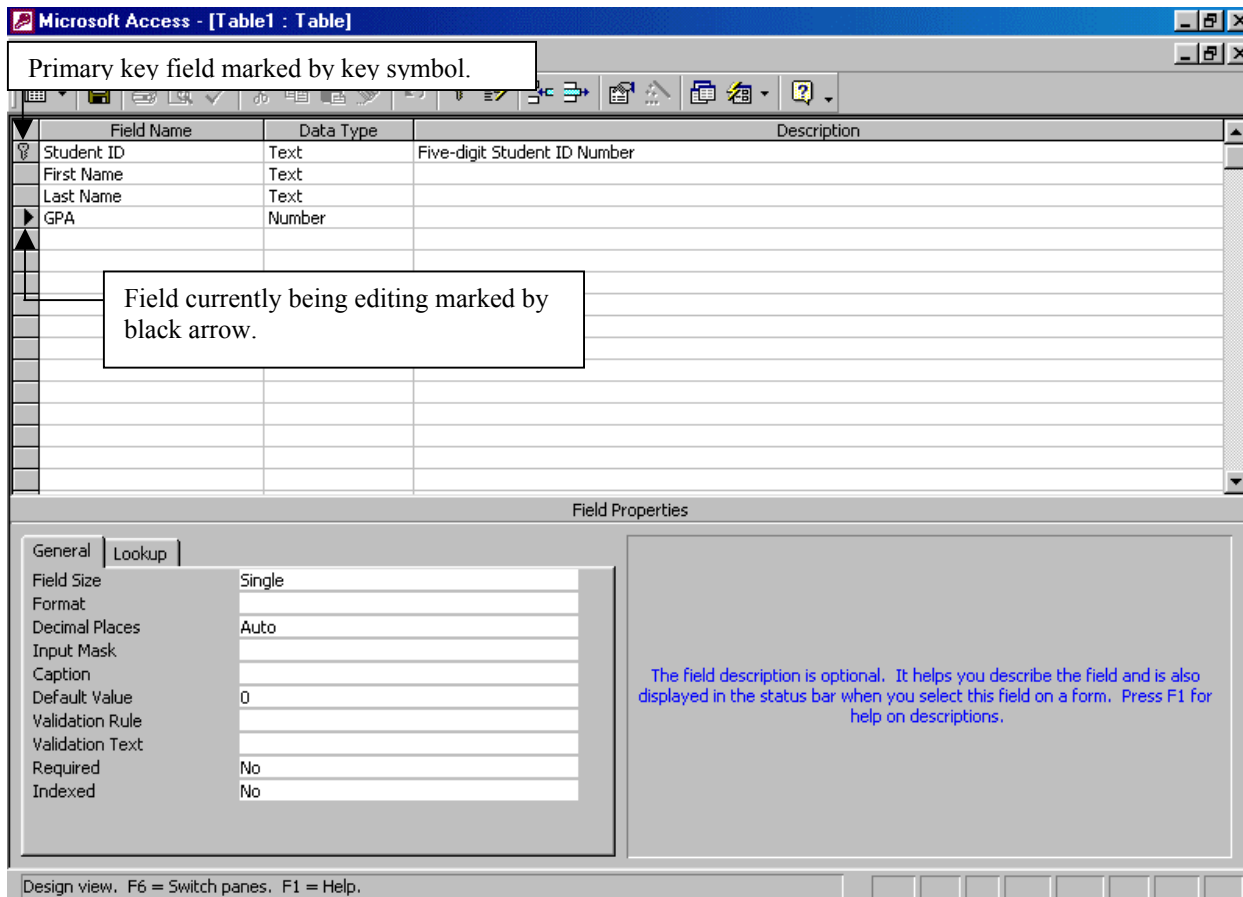
Data Type	Explanation
Text	Plain text, including letters, numbers, etc. The maximum field length is 255 characters.
Memo	Also text, with a maximum length of 65,535 characters. Be wary of using--it can take up a lot of disk space!
Number	Numbers only--no text. Check the help available within the Field Size in the Field Properties box for tips on defining the size.
Date/Time	A date or time, with several options in format (short date, long date, etc.) available via a pull-down menu that will be displayed under Format in the Field Properties.
Currency	Numbers will be automatically converted to currency. Check the Format box in the Field Properties for options.
AutoNumber	Access automatically fills in a consecutive number for each record added. Good for those times when you want a Primary Key (see below) but can't think of an existing field to assign one to.
Yes/No	For a field that requires a Yes/No, True/False, or On/Off answer.
OLE Object	A link to a file or database; too advanced for the purposes of this handout!
Hyperlink	To create a link to a file on the Web; your database must on a computer with constant connection to the World Wide Web for this to work.
Lookup Wizard	Allows one to choose values listed from another table; too advanced for the purposes of this handout.

Primary Key


Before saving your table structure, it is generally a good idea to assign a **Primary Key** to a field. A field that contains unique information about an individual (for example, a Social Security Number or a student ID number) is a good candidate for a primary key. A primary key field means you will *not* be able to have records with duplicate entries in that field. A primary key is not a requirement, though *Access* will ask you if you want one before you save your table, and if you answer **Yes** (to assign one), it will create a new field on its own, generically called ID and given the data type of AutoNumber. Therefore, it is best to pick a field you have already created. To assign a primary key, place your cursor anywhere in the row representing the field you want to make the primary key, and click on the **Primary Key** button on the table design toolbar: 

Completed Table

Here is our table after adding four fields to it:



Saving Your Table Structure

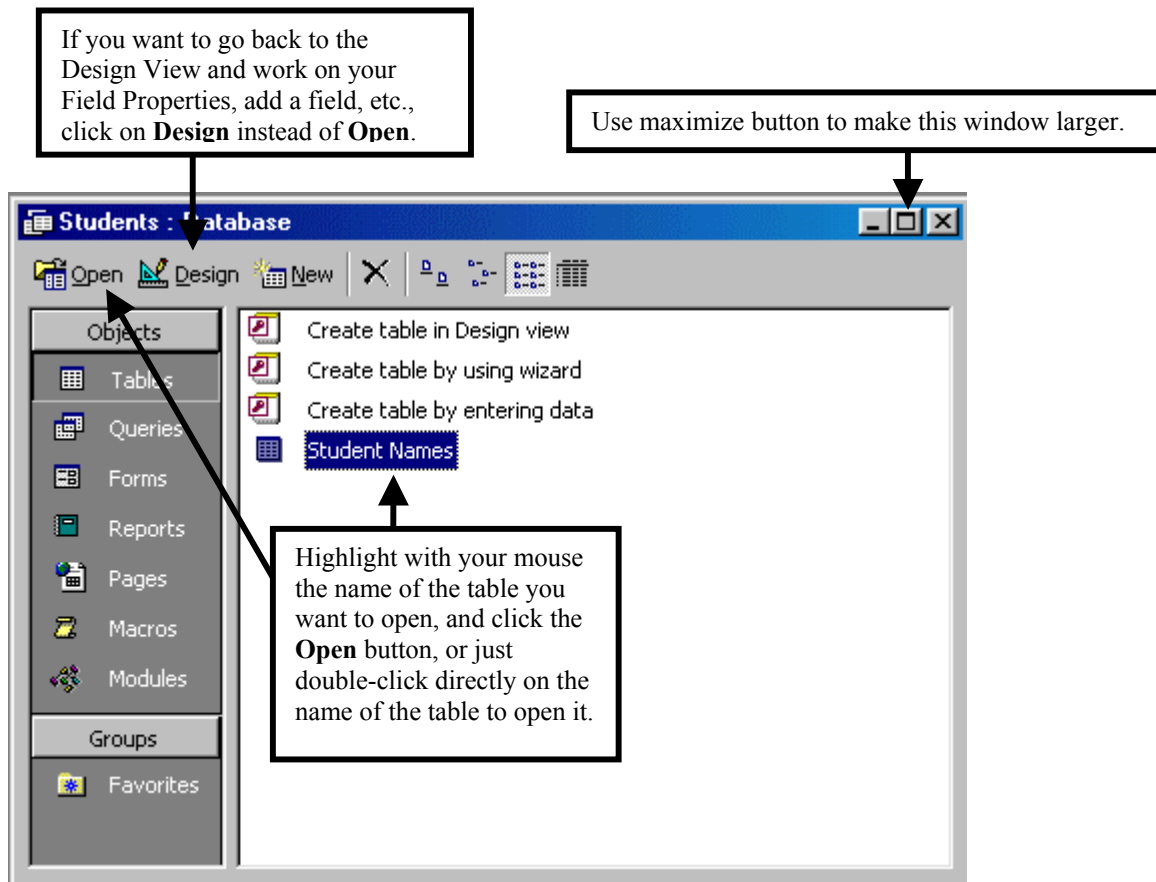
Once the table structure has been defined, the next step is to save the table within the database. (Keep in mind a single database can have many tables.) To save a table, click on the **File** pull-down menu, then **Save**, or just click on the floppy disk icon on the toolbar:  You will then see the following dialog box. After typing in a name, just click **OK**:



One naming convention often used is to precede your table name with the letters **tbl**. So, the name you save your table under could be **tblStudent Names**. Even more standard is to eliminate spaces altogether, which can make it easier if you ever need to use your table in another program: **tblStudentNames**.

ADDING RECORDS AND EDITING RECORDS IN A TABLE

Once the fields for your table have been defined, you are ready to put actual records within the table. In order to add records to a table, the table must be open. First, close the design view (if it isn't already), so that you're back to the main database window in *Access*:



When you open your table, you will be in what is called the **Datasheet View**, and your table will be ready to have records added to it. The easiest way to add records in this view is to TAB your way through each of the fields. The screenshot on the next page shows the datasheet view. Note the "move" buttons at the bottom of the screen that can be used to navigate from record to record--particularly as your database grows in size.

Microsoft Access - [Student Names : Table]

File Edit View Insert Format Records Tools Window Help

Student ID	First Name	Last Name	GPA
12345	Samuel	Adams	3.45
98765	Teddy	Bare	2.98
33224	Barbie	Dahl	3.23
44332	John	Doe	2.99
*			0

Record: 4 of 4

Datasheet View

Go to first record

Previous record


Next record

Go to last record


Move to end of database to create a new record

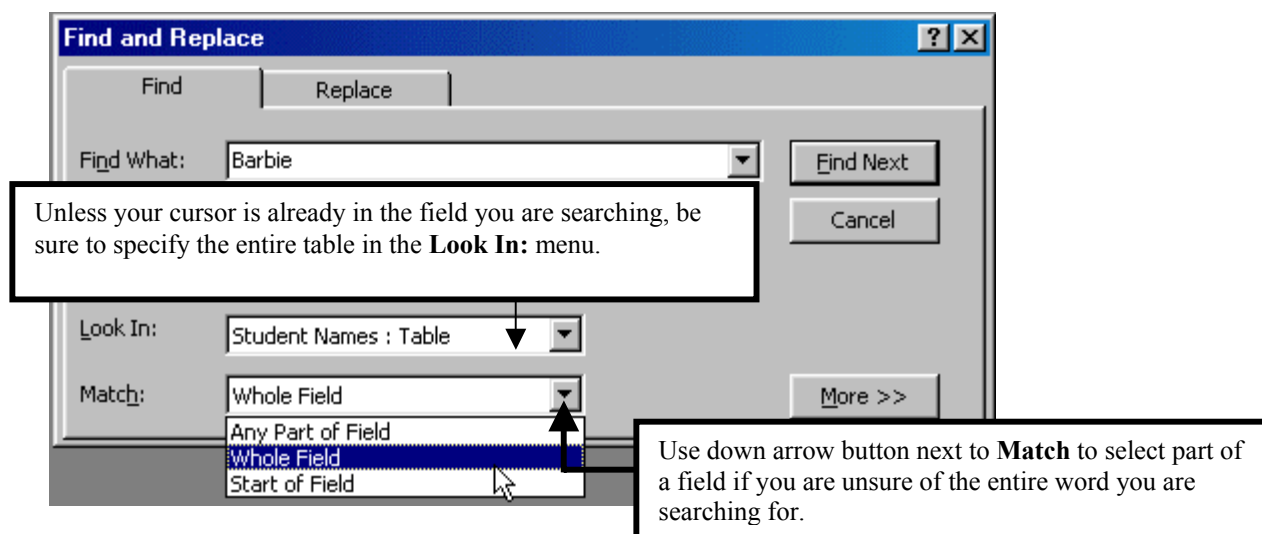
Click this button to go back Design View.

Press TAB key on the keyboard to move insertion point through the fields after you've typed your data, or click within the field the mouse.

To add a new record to your table in the Datasheet View, just click on the  (Go to last record) button near the status bar on the bottom of the screen. To *modify* information already in your table, just click in the appropriate box and enter the new value. Unlike other *Office 2000* applications, *Access* saves new record information automatically--you do not even have to click the Save button on the toolbar. However, if you change anything in the database structure in the Design View (length of fields, a new field, etc.) you *must* save it!

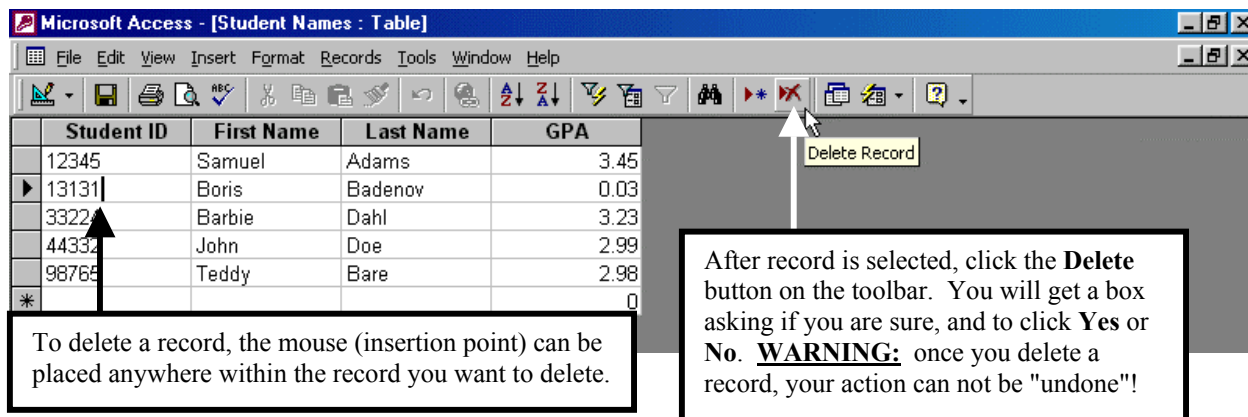
Finding a Record

If your table gets big (and most active database tables will!) and you need to find a specific record to make changes to it, there are several ways to do this, many beyond the scope of this document. One quick way, however, is to simply click on the **Find** button () in the toolbar and type in the text you are looking for. *Access* will take you to the record(s) that match your criteria:



DELETING RECORDS FROM A TABLE

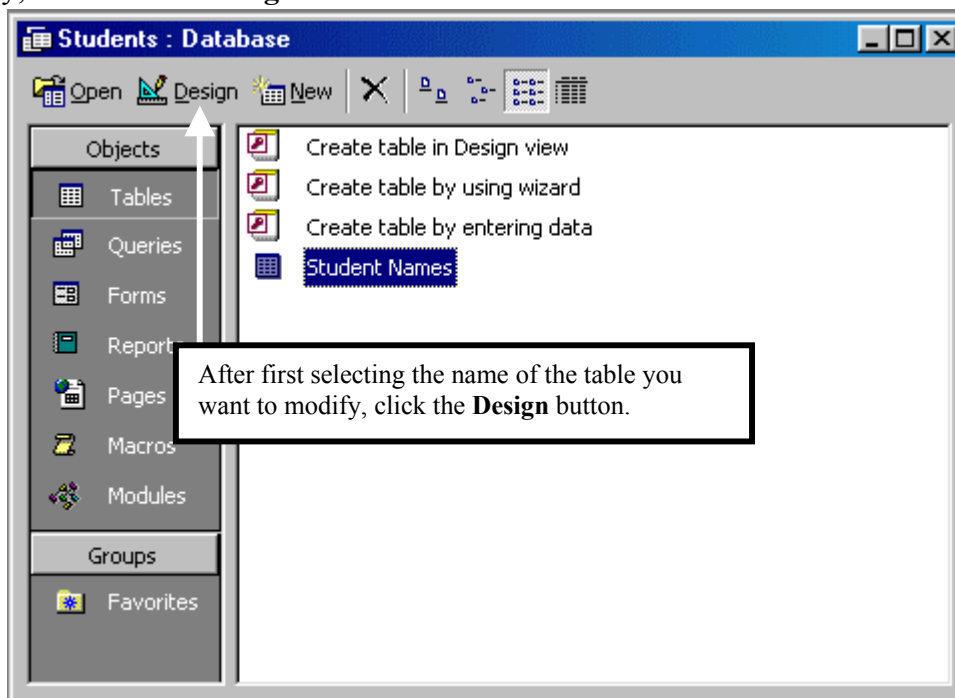
To delete a record, you must find the exact record you wish to delete. (You may want to use the **Find** feature outlined above.) Once you have found it, click *anywhere within* that record. Then, click the **Delete** button on the toolbar:



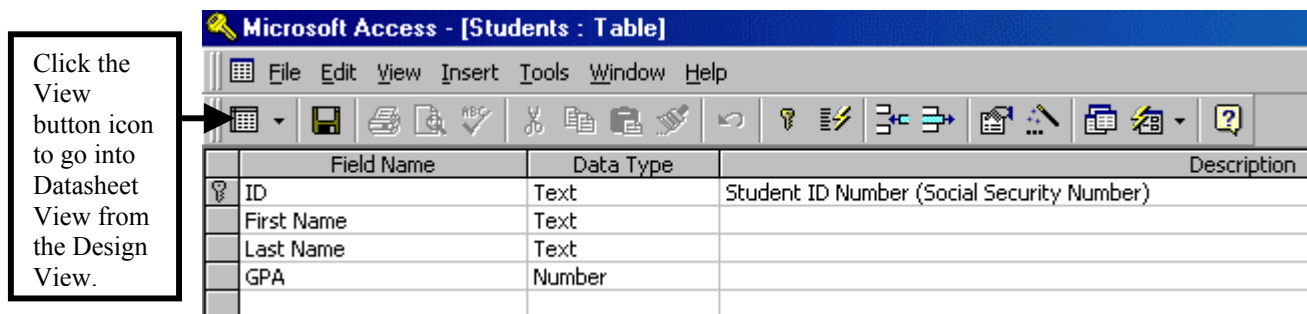
MODIFYING A TABLE

It is possible to modify a previously saved table's structure, though you should be aware that there could be several consequences. If you decide to add a field, you will have to go through each record in your database table and add the data for your new field. (Fortunately, you can perform an *update query*--discussed later in this handout--to do this task.) If you change the field size property, you may lose data if you shorten it too much (though *Access* gives you fair warning first).


To change the design of a table, when you first open your database, make sure the **Tables** tab is clicked in the main database window, click once over the name of the table you want to modify, and click the **Design** button:



If you are already in the table Datasheet View, click the **View** button (see screen shot on page 10) to go into the Design View. Alternatively, if you are in the Design View and want to go back to the Datasheet View, you may also click the **View** button, which will have a different icon this time:

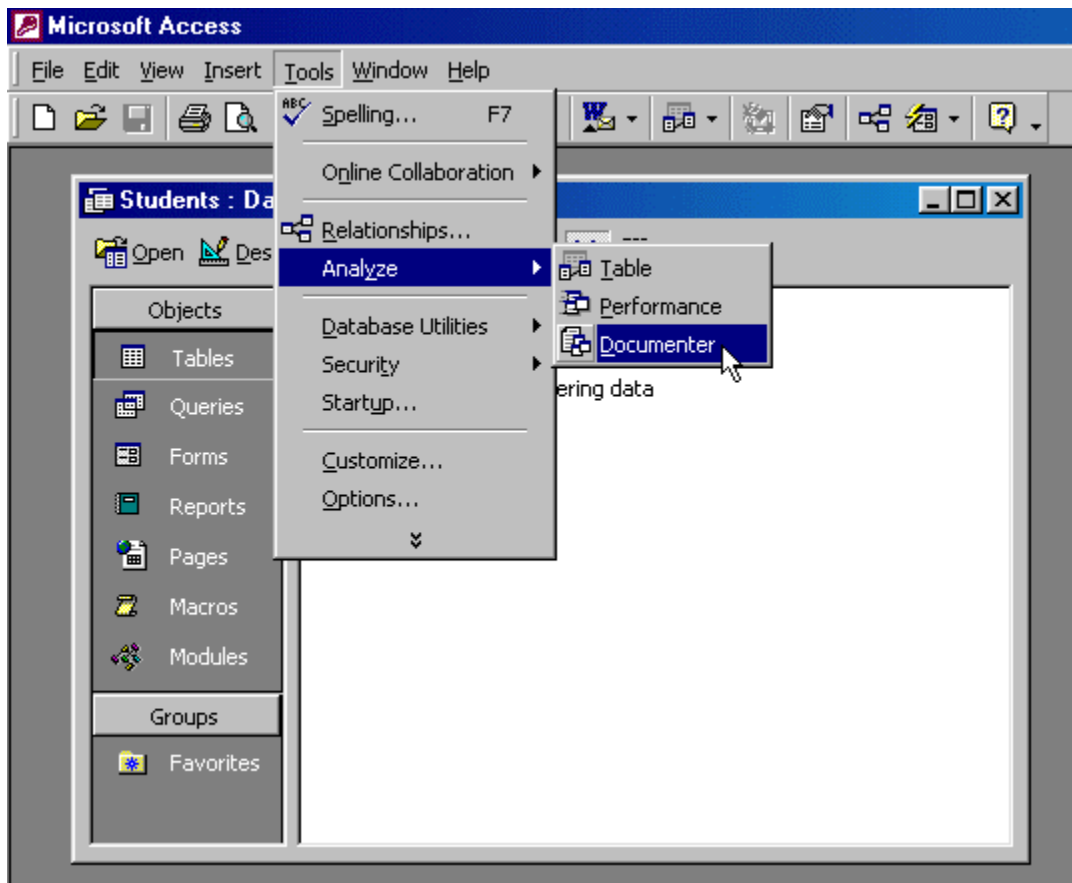


PRINTING A DATABASE TABLE

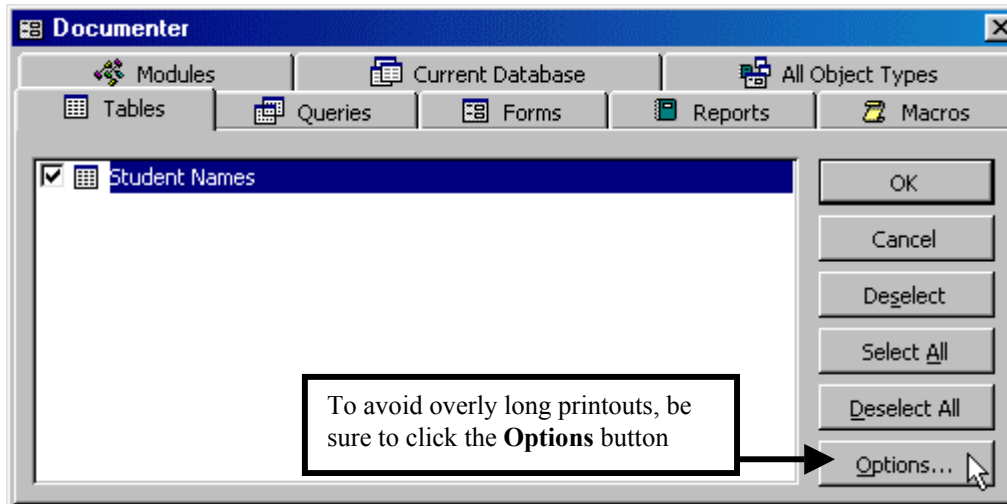
To print a database table, just click on the **File** pull-down menu, then **Print**, as you would for any other *Office 2000* application. Creating a Report, discussed later, provides a more eye-pleasing output, however. You may want to click the **Print Preview** button on the toolbar to see how your table will look before printing it: 

PRINTING THE TABLE STRUCTURE

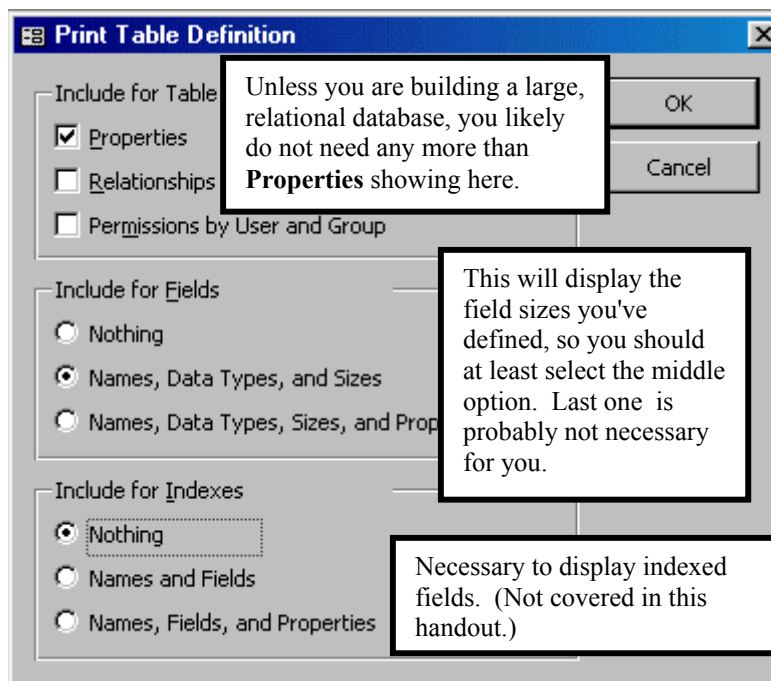
There may be times you want to view the structure of your database all at once on a printout without having to actually go into the database. *Access* allows this using a tool called the Documenter. To get to the Documenter, you must first *close* the tables you are editing and be at the main database window. From here, you want to click the **Tools** pull-down menu, then **Analyze**, and finally **Documenter**:



Once you do this, you will be presented with the following dialog box:



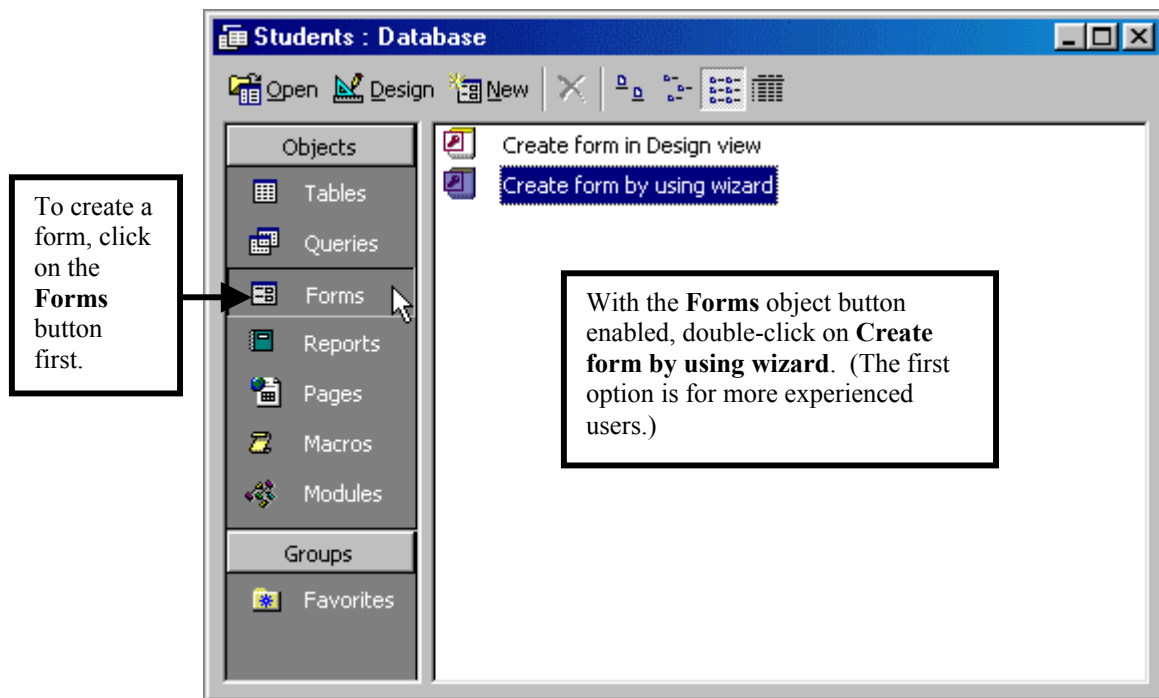
After clicking next to the table name you want documented, before going any further (and to avoid multi-paged printouts, depending on the part of the database you are documenting), you should click on the **Options** box in this dialog box to pick out the parts you want a printout for:



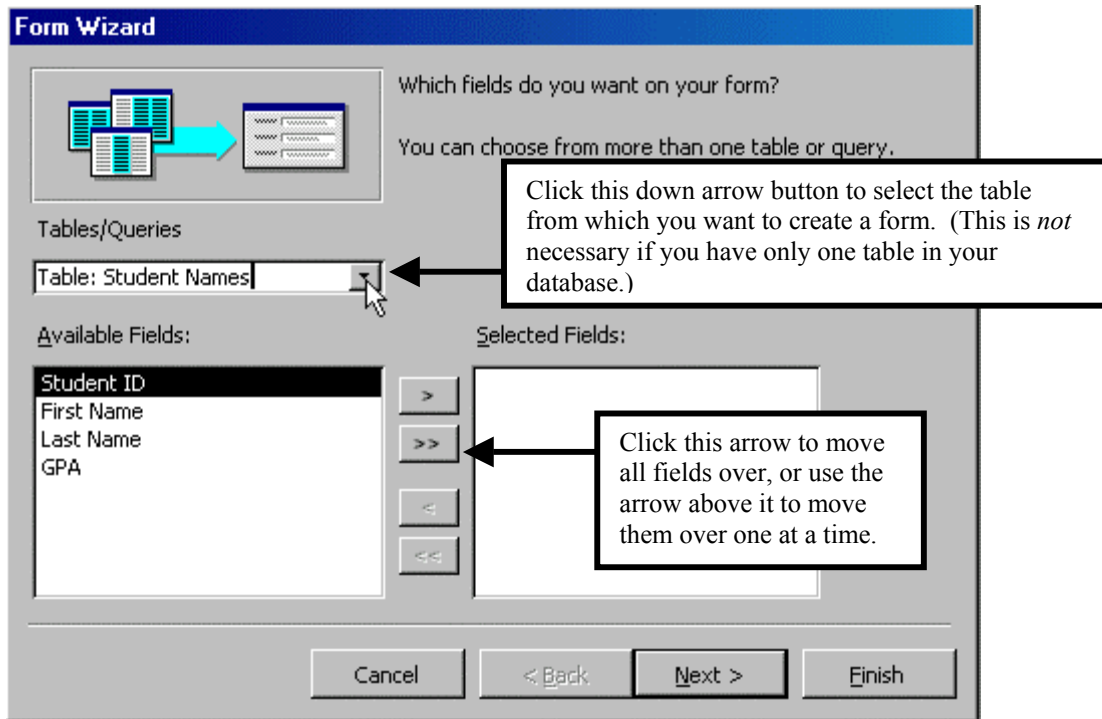
After you have made your selections, click **OK**. You will return to the previous dialog box (shown above). At this point, you can click **OK** and you will be shown a preview of your printout. You may printout this screen using the usual **File . . . Print . . .** command.

CREATING A FORM TO INPUT DATA

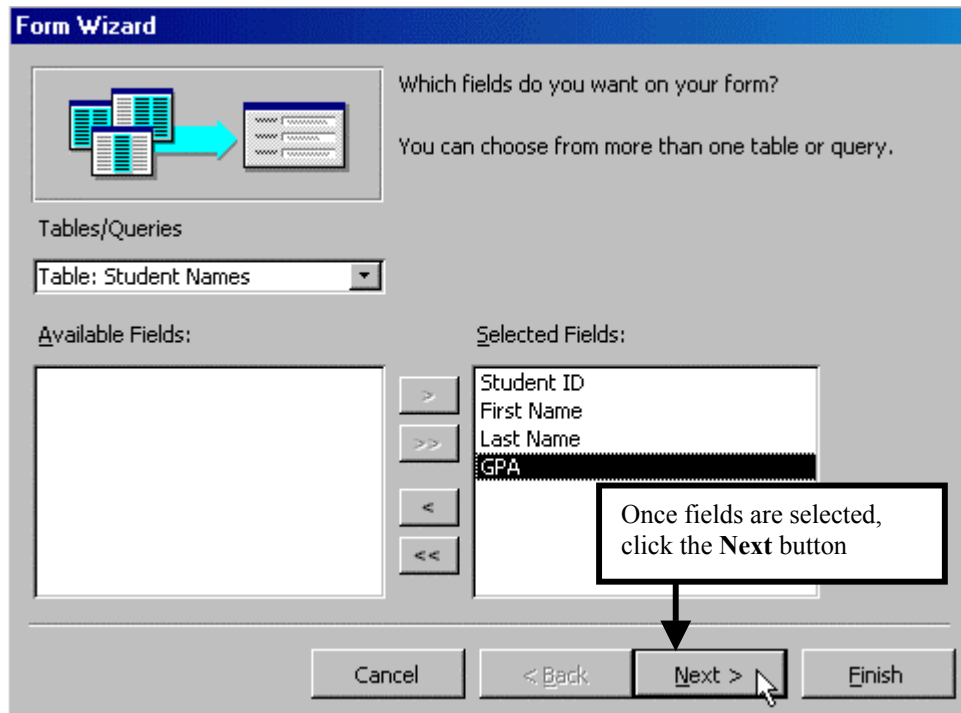
If you have ever input records into *Access* while working in an office, you likely did so through a *form*. Forms typically provide a much easier layout for inputting records than the Datasheet View does. This document cannot go into too much detail on creating forms in *Access*, but we can provide a quick way to create at least a rudimentary form for data input. From the main *Access* database window, make sure you first click on the **Forms** object button on the left, and when enabled, choose **Create form by using wizard**:



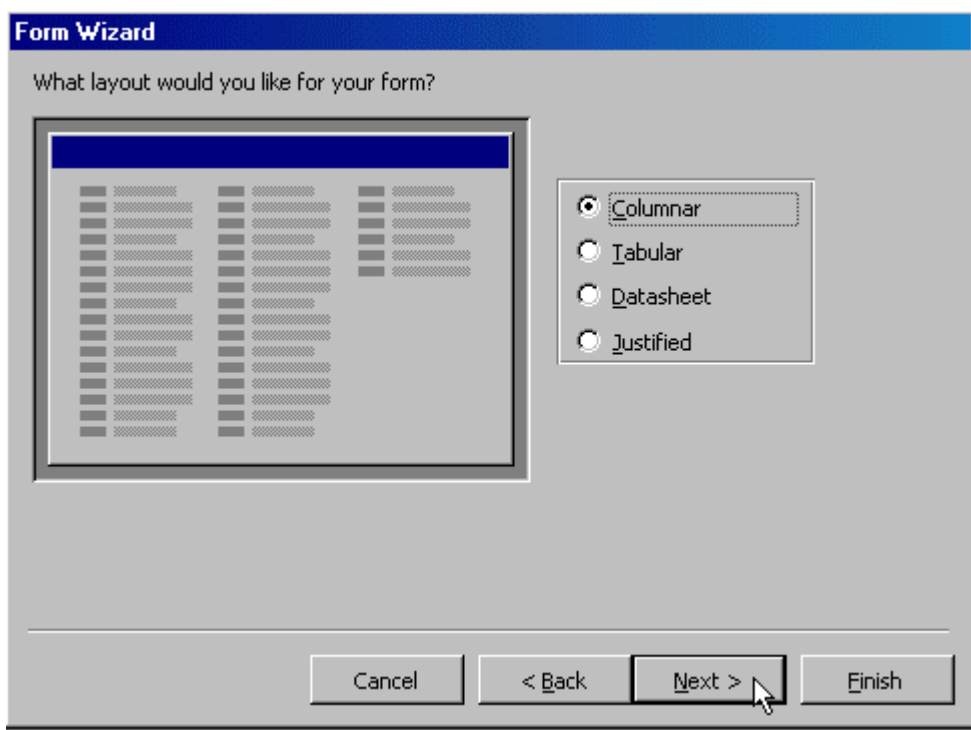
This will lead to the following dialog box shown on the next page. The Form Wizard will guide you step-by-step through the form creation process. The first box prompts you for which fields you want to include on the form, and from which database table. You can select them one-by-one by highlighting the name of the field and then clicking the > button, but if you want all fields on the form (and you likely will), just click on the >> button to move them all over at once:



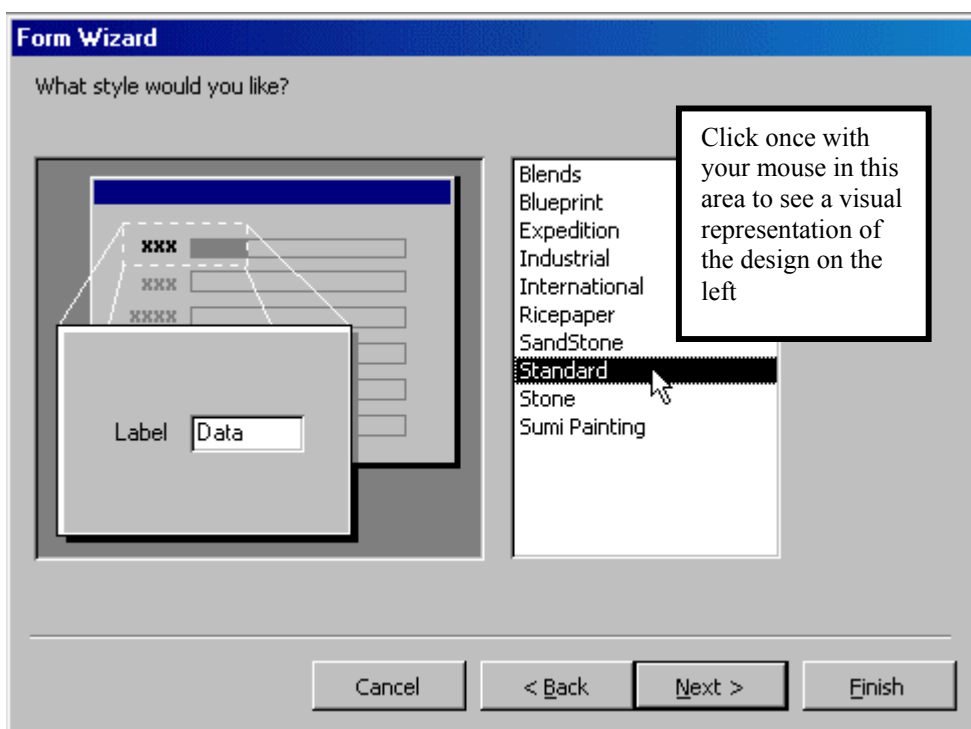
Once you have selected your fields, click the **Next** button.



You are next asked to select the appearance of your form. Keep your eye on the window on the right for an idea of the layout. (The **Tabular** and **Datasheet** views somewhat resemble the **Datasheet View** of the table, while columnar forms are arranged vertically).



You are next prompted to select the visual background and appearance of your form:



Finally, you will be asked to name your form. By default, *Access* will have the same name as your table already in the box. You may want to consider giving it a slightly different name, since there may be times while using *Access* that you will have to select a table, query, or form name from a drop-down menu listing, and too many elements in *Access* with the same name can be confusing. A commonly used naming convention is to simply place the letters **rpt** in front of the name *Access* provides:

Access will then display the completed form. You may now add records using the form rather than the table Datasheet View. (There are many, many more possibilities in *Access* with designing a form than this overview has the chance to give you. Consult a book on *Access* for more information.) On the next page is a screenshot of the completed form.

FORM VIEW

Microsoft Access - [rptStudent Names]

File Edit View Insert Format Records Tools Window Help

MS Sans Serif 8 B I U

Student ID: 12345
 First Name: Samuel
 Last Name: Adams
 GPA: 3.45

Remember the optional **Description** field in the Table Design View? That description appears at the bottom of the screen in both the Form view and the Datasheet View

While adding data in a form, you may press the TAB key to move from field to field as you did in the Datasheet View. Also, note that you can move between records using the same controls at the bottom of the screen that were seen earlier in the Datasheet View

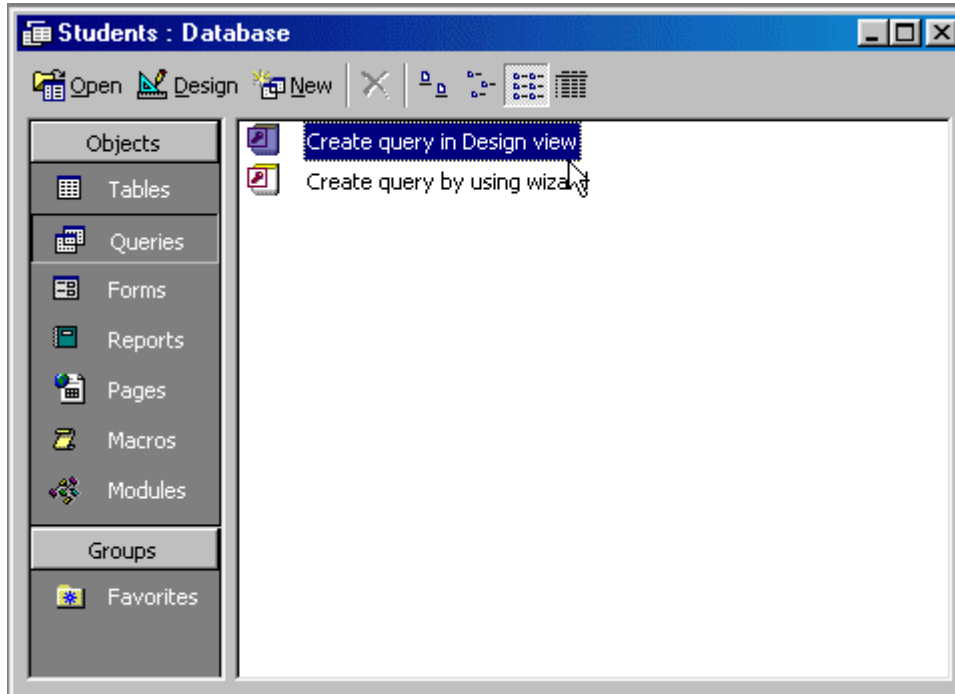
Rec'd: 1 of 5
 Five-digit Student ID Number

QUERIES

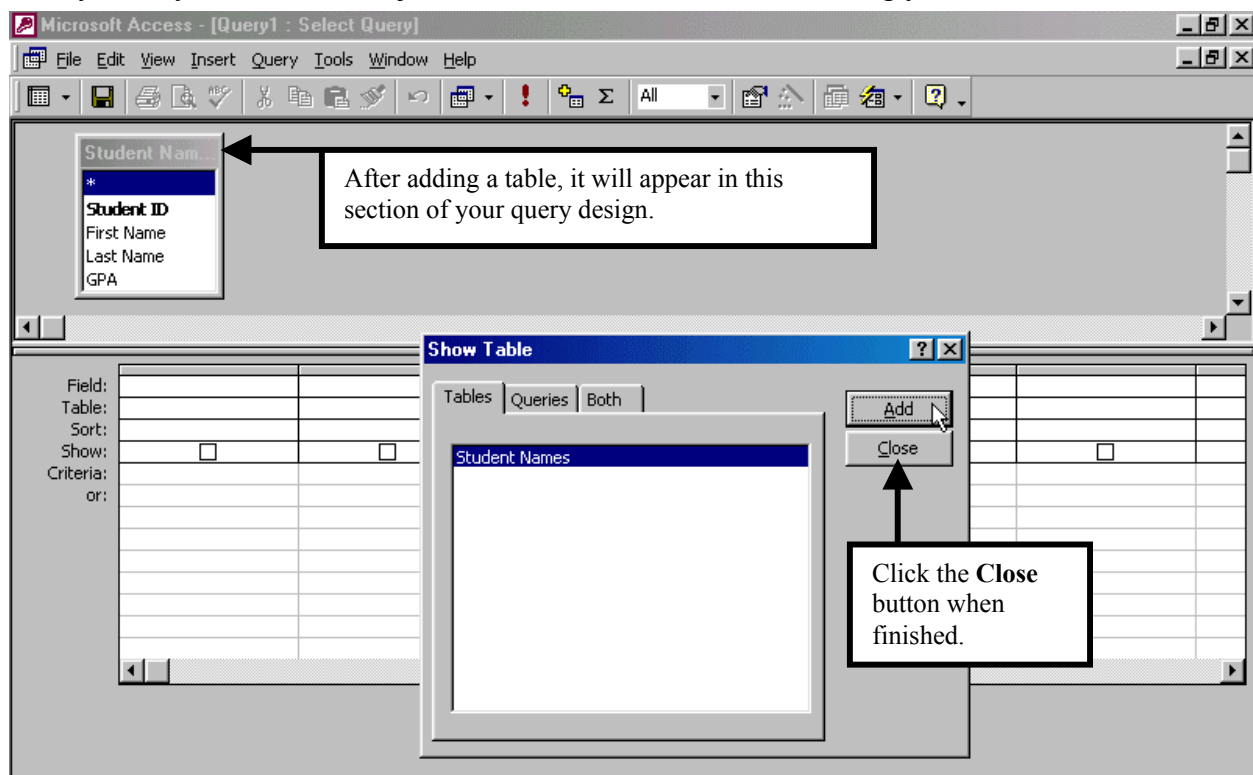
As your database gets bigger, it gets more difficult to find a specific record or records that match certain criteria. You may, for example, want to know the names of all students who have a grade point average of 3.0 or above. When you ask a question of *Access* or any other database, that question is called a query. To find an answer to a question in a database, you must first create a query, and then run the query. Queries can be saved, so that if you have to keep asking the database the same question, you can keep running the query to get the results—even if you've added hundreds of records to the database after you first constructed the query.

Creating a Query

To create a new query, make sure the main database window is open, and from there, click on the **Query** object, then from the resulting list, double click **Create query in Design view**.

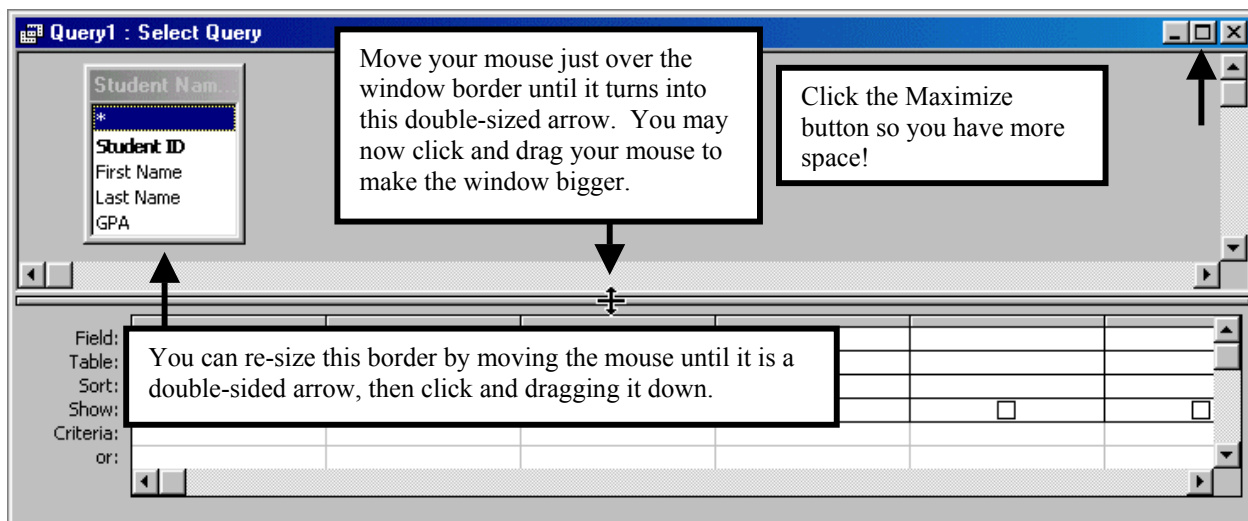


Next, you will be asked which table(s) to have in your query design. Highlight the table name, then click **Add**. You'll notice a table appearing in the upper part of the query design window. Since you only have one table, just click the **Close** button after adding your table.



Using the Query Window

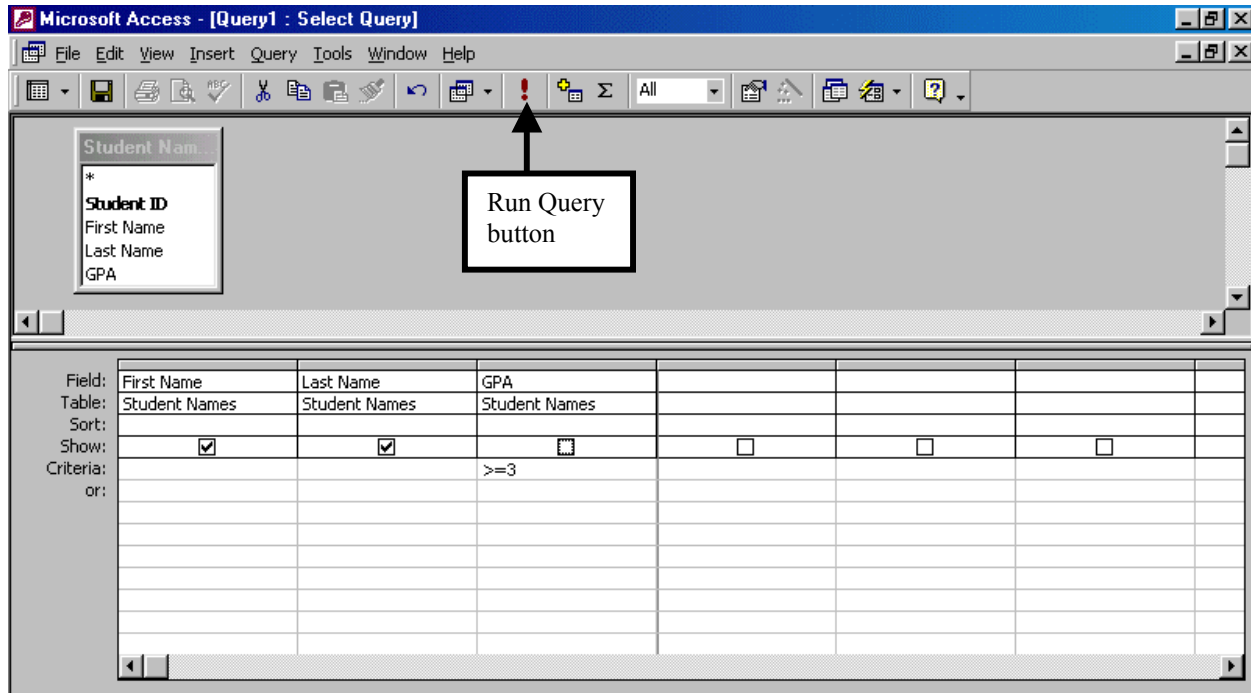
You will now be in the Query Design View. At this point, you may find it easier to maximize the window if it is not already. In addition, you can use your mouse to move the upper portion of the window higher or lower in relation to the lower portion. Also, click and drag over the edge of the table window in the upper window if you cannot see all of the fields in the table. You will need to be referring to these fields when constructing your queries.



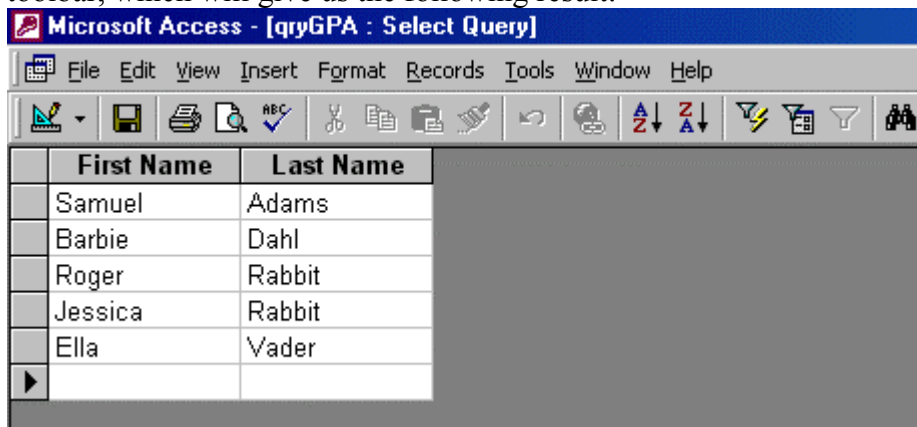
The purpose of the query design window is to specify the fields you want to run a query on. There are a few ways to do this:


- Double-click on the name of any field from the table window in the upper part of the window, or
- Click on the name of any field, then while holding down the left mouse button, drag it down into the lower window, or
- Click in the **Field** row on the bottom half. A down arrow will be displayed. You can click on this down arrow to select a field.

There will be specific query examples discussed below. You will also notice there is a row labeled **Show** in the lower window. If the Show box has a check mark (☑) in it, it means the field will be displayed in your resulting query. If there is no check mark, the field will not be displayed. Why run a query if you do not even want a field displayed? One example may be that you want a listing of all students with GPAs of 3.0 or higher. You will have to specify the **First Name**, **Last Name**, and **GPA** fields in the lower window to run the query, but you may only want to **show** the first name and last name, and not the actual GPA in your results:



The query above shows that we want to show the first name and last name of students. We want only those students listed with a GPA of 3.0 or higher, so in the **Criteria** row we typed in ≥ 3 . (There is no need to add the ".0" after the 3 in this case.) When we actually execute the query, however, we do not want the actual GPA displayed, so we unchecked the **Show** box in the GPA column. Now, to run the query, we need only click on the exclamation point button in the toolbar, which will give us the following result:



This result shows first name and last name, but not the GPA. Of the five students in our database so far, these are all of those with GPAs of 3.0 or higher. If you did not get the results you expected, you can go back to the design view by clicking the view button: 

QUERY EXAMPLES

There are many types of queries that can be run in *Access*. This section will outline a few. Before going any further, however, have a look at our slightly enhanced database, so that you can follow along with what the results should be when looking at our examples. Notice that we have added a field (Home State) and a few more students:

Student ID	First Name	Last Name	GPA	Home State
12345	Samuel	Adams	3.45	IL
13131	Boris	Badenov	0.03	WI
22222	Roger	Rabbit	3.72	IN
32154	Hope	Springseternal	1.75	IN
33224	Barbie	Dahl	3.23	IL
33333	Jessica	Rabbit	3.50	IN
44332	John	Doe	2.99	WI
55555	Ella	Vader	4.00	WI
98765	Teddy	Bare	2.98	IL
98989	Dewey	Cheatum	2.01	IL

Here are the way the fields were defined in the Table Design View:

Field Name	Data Type	Length	Other Pertinent Information
Student ID	Text	5	Primary Key
First Name	Text	15	
Last Name	Text	20	
GPA	Number	Single	In Field Properties, following were set: Field Size: Single Format: Fixed Decimal Places: 2
Home State	Text	2	

Greater Than or Less Than

You may use the following in the **Criteria** row of an *Access* query to match numbers or letters greater than or less than specific criteria:

> greater than	< less than	= equal to	<> not equal to	>= greater than or equal to	<= less than or equal to
----------------	-------------	------------	-----------------	-----------------------------	--------------------------

This works with any date fields you may have, too. So, to find all students whose last names begin with a letter greater than the letter **L** and who have GPAs of less than 3.25, you would enter:

Field:	First Name	Last Name	GPA
Table:	Student Names	Student Names	Student Names
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		>"L"	<3.25
or:			

Access will add the quotation marks automatically around certain criteria-- you do not have to enter them yourself.

This will yield the following result when the **Run Query** button is pressed:

First Name	Last Name	GPA
Hope	Springseternal	1.75
		0.00

Using Wildcards

The asterisk (*) is the wildcard symbol in *Access*. It is handy if you are unsure of the exact spelling of a word in your database or simply want everything starting with a certain character string. To retrieve all students whose last name begins with the letter **D**, for example, you would enter **d*** in the **Criteria** field (upper case letters are not necessary):

Field:	First Name	Last Name	
Table:	Student Names	Student Names	
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Criteria:		Like "d*"	
or:			

Access automatically adds the word "Like" before any criteria using the wildcard character, and encloses your criteria within quotation marks.

Here are the results from this query:

	First Name	Last Name
	Barbie	Dahl
	John	Doe

Logical Operators

You may use the logical operators AND, OR, or NOT within queries, though *Access* does some of this automatically within the query design window. For example, each column in the query is an implied AND--you want the contents of one field AND the contents of another to appear in the results. Our first example used a logical AND--we wanted students with last names after the letter **L**, AND we wanted to limit the results to those with a GPA less than 3.25.

The logical operator OR may be typed directly between two criteria in the same cell or used the next line down in a column (note from our previous screen shots that the **Criteria** row has a row labeled **or:** just beneath it). When using OR, it means you want one OR the other to appear in the results. Therefore, if we want students whose first name is Barbie or Boris, we can enter it as:

Field:	First Name	Last Name	
Table:	Student Names	Student Names	
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Criteria:	"barbie"		
or:	"boris"		

Typing an OR between the two terms in the same Criteria row can perform the same search.

Field:	First Name	Last Name	
Table:	Student Names	Student Names	
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Criteria:	"barbie" Or "boris"		
or:			

In fact, this second method is *preferable* for more complex queries with several columns of criteria. There is an implied AND between columns, but also across *each* row. Here is an example. Say we want to search for students named Barbie or Boris, but those students must also have a GPA of at least 2.5. Putting this search into the Query Design View using our first method (the "OR"s down one column), would look like this:

Field:	First Name	Last Name	GPA
Table:	Student Names	Student Names	Student Names
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	"barbie"		>=2.5
or:	"boris"		

Note, however, the result this query retrieves:



	First Name	Last Name	GPA
	Barbie	Dahl	3.23
	Boris	Badenov	0.03
			0.00

This clearly does not match the intent of our query--since Boris has a GPA far below our requested GPA of 2.5 or above! Here is what happened. *Access* looked at our first row of criteria--student First Name of Barbie and the implied AND of a GPA greater than or equal to 2.5. However, because we decided to use the **or:** row below the First Name field, *Access* then looks at the criteria in *that* row (student First Name of Boris) and . . . nothing else--since there is no criteria for that row in the GPA column:

Remember there is an implied "AND" between columns and across each row. But, if there is more than one row under **Criteria:**, *Access* assumes you want the results of one row OR the results of the other. One potential solution to this specific example is to re-type the >=2.5 criteria a second time in the GPA column under the first one. Easiest, however, would be to manually type in OR between Barbie & Boris in the first row.

Field:	First Name	Last Name	GPA
Table:	Student Names	Student Names	Student Names
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	"barbie"		>=2.5
or:	"boris"		

To avoid this problem, it is best to use the preferred method of simply typing an **OR** between the criteria on the same row as your other criteria:

Field:	First Name	Last Name	GPA
Table:	Student Names	Student Names	Student Names
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	"barbie" Or "boris"		>=2.5
or:			

Access automatically puts our criteria within quotation marks when we type in an **OR** between two terms. Also, remember the query is *not* case sensitive.

Using this method returns the results we expect:

	First Name	Last Name	GPA
	Barbie	Dahl	3.23
			0.00

We get just one result, as we should, since Barbie is the only student named "Barbie" or "Boris" who *also* has a GPA of 2.5 or above.

The NOT operator is used by placing the word NOT *before* the criteria you do not want to appear in your query. If you want a list of all students, but not those whose last name begins with the letter **D**, you would enter **not d*** in the **Criteria** row under Last Name. (*Access* automatically adds the word *Like* as it does for all wildcard queries with the asterisk, and also encloses the **d** within quotation marks--you do not *have* to type in either):

Field:	First Name	Last Name
Table:	Student Names	Student Names
Sort:		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		Not Like "d*"
or:		

Here are the results of this query:

	First Name	Last Name
▶	Samuel	Adams
	Teddy	Bare
	Boris	Badenov
	Roger	Rabbit
	Hope	Springseternal
	Jessica	Rabbit
	Ella	Vader
	Dewey	Cheatum
*		

BETWEEN Operator

The BETWEEN logical operator is good for searching between two values, inclusive. For example, you may want to search for all students with a GPA between 2.00 and 2.99, inclusive. In the GPA field, you should enter **between 2.00 and 2.99** in the **Criteria** field:

Field:	First Name	Last Name	GPA
Table:	Student Names	Student Names	Student Names
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			Between 2 And 2.99

(Note that *Access* just strips off the zeroes following the decimals). The same query could have been constructed using the \geq sign along with the \leq sign. This is just one example of how the same query can be constructed several different ways:

Field:	First Name	Last Name	GPA
Table:	Student Names	Student Names	Student Names
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			≥ 2 And ≤ 3

Either query gets the same results when the **Run** button is clicked:

	First Name	Last Name	GPA
	Teddy	Bare	2.98
	John	Doe	2.99
	Dewey	Cheatum	2.01
▶			0.00

Sums, Averages, Etc.

Clicking on the sigma key on the toolbar in the *Access* query design window (Σ) will bring up a new row called **Total** in the query grid. This allows you to do some basic arithmetic operations, such as summing certain fields, giving the average, etc. Clicking the down arrow within this row will give you a list of the capabilities. For example, to compute the average GPA among the students in the database, you would include the GPA field in your query, and select **Avg** after clicking the drop-down arrow:

Field:	GPA	
Table:	Student Names	
Total:	Avg	▼
Sort:		
Show:	<input checked="" type="checkbox"/>	
Criteria:		
or:		

After running the query, the result will simply be the number:

	AvgOfGPA
▶	2.76600001137704

Be careful not to add fields that cannot be displayed together when doing a search with the **Total** row. You cannot, for example, include the Last Name field in the above query, because all you want is an average. If you try, *Access* will simply list all of the last names in your database, and not bother trying to calculate the average at all.

You will find, when combining certain fields you wish to total, that you may have to specify the term **EXPRESSION** in the **Total** row to get accurate results for your query. For example, say you want the average GPA of all students with last names greater than the letter **L**. Here is the way you may want to start your query:

Field:	Last Name	GPA	
Table:	Student Names	Student Names	
Total:	Group By	Avg	
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Criteria:	>"L"		
or:			

This looks logical, but *Access* will in fact give you query results that look like this:

	Last Name	AvgOfGPA
	Rabbit	3.61000001431
	Springseternal	1.75
▶	Vader	4

Instead of displaying the average GPA for the students whose last name is greater than L, *Access* instead simply lists them. To further confuse matters, because there happens to be two students with the last name Rabbit, *their* GPA is averaged.

Basically, if combining a **Total** row with other fields, depending on your criteria you may have to specify the term **EXPRESSION** within the **Total** row in the other columns. (Think of it this way: you want the average GPA of all students higher than the last name "L," but you are including this as part of your total *expression*.) The default setting (as seen in the screenshot before this one) is **Group By**, which means *Access* will *group* the results *by* Last Name. Consequently, when two individuals have the same last name, they constitute their own grouping, and *Access* calculates their average. In addition, you will want to *uncheck* the **Show** box in the Last Name column, since you do not want the names to show on the result. Here is the way the correct query design should look:

Field:	Last Name	GPA
Table:	Student Names	Student Names
Total:	Expression	Avg
Sort:		
Show:	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	>"L"	
or:		

Now, when you run it, you get just one average--as you should:

	AvgOfGPA
▶	3.24250000715256

Renaming a Field

Just because you named a field one way in your table design doesn't mean that's the way it has to appear in query results. To display a field in a query using a different name than the one you originally gave it, click in the field name in the bottom portion of your query design, and simply type the name you want to give you field **in front of the original name**, followed by a colon (:) with no spaces separating your name from the original name. The format is:

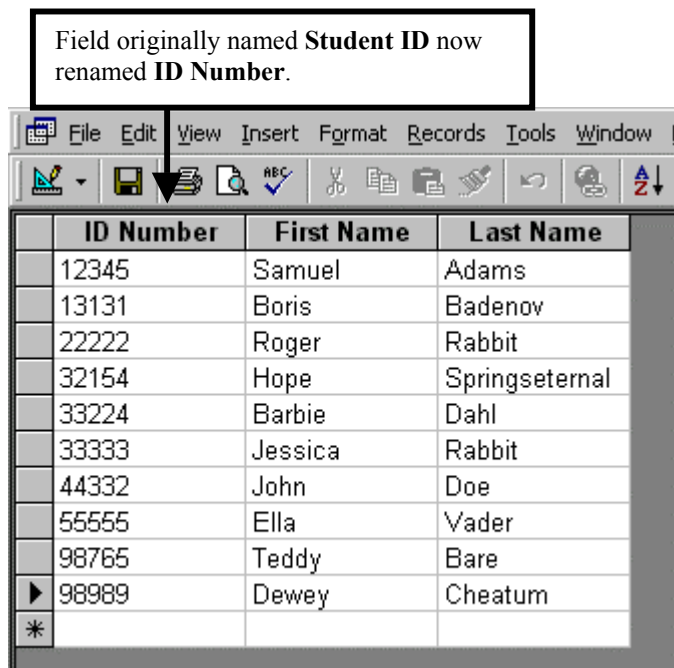
NEW FIELD NAME:OLD FIELD NAME

So, to rename the field we originally called **ID** to one called **Soc Security Number**, here is what our query design would look like:

Field:	ID Number: Student ID	First Name	Last Name
Table:	Student Names	Student Names	Student Names
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			
or:			

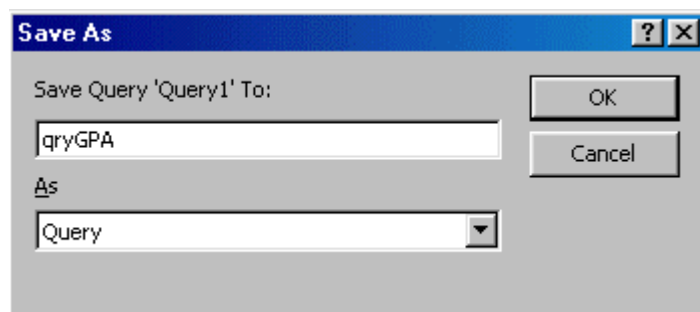
Column widths can be changed by positioning the mouse between columns and dragging.

You can type directly into any box in the query design table, but you can also press **Shift-F2** to bring up a "zoom" window once your cursor is in the correct field name in the query, which gives you a bigger space to work in. Just make sure to keep your original field name after the colon! Here is the result of this query with the renamed field:



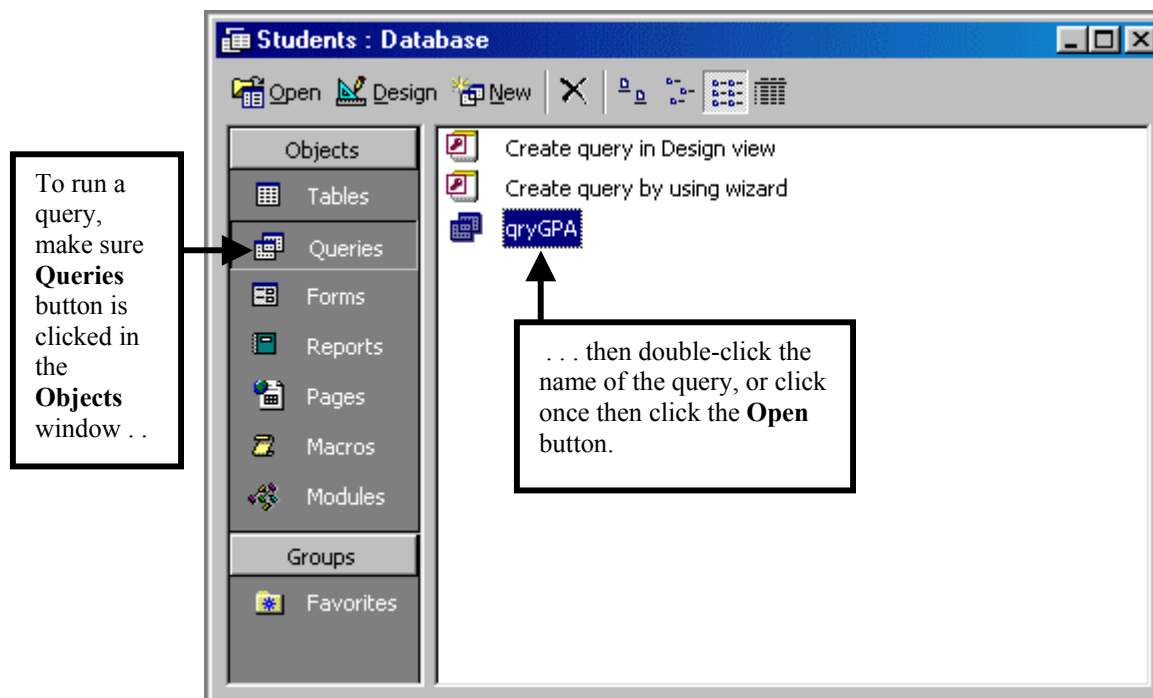
SAVING A QUERY

After you have designed a query and run it, and everything looks like it works, you should immediately save it. (After all, why do it all over again?) To save a query, in either the design view or the results view, click on the **File** pull-down menu, then **Save** if it's the first time you're saving the query, or **Save As** if you made modifications to it but don't want to over-write your first version. In the resulting screen, you can give your query a name. By default, *Access* names them **Query1**, **Query2**, etc., but you can change it. One popular naming convention is to precede the query name with the letters **qry** followed by the name of the query:



RUNNING A SAVED QUERY

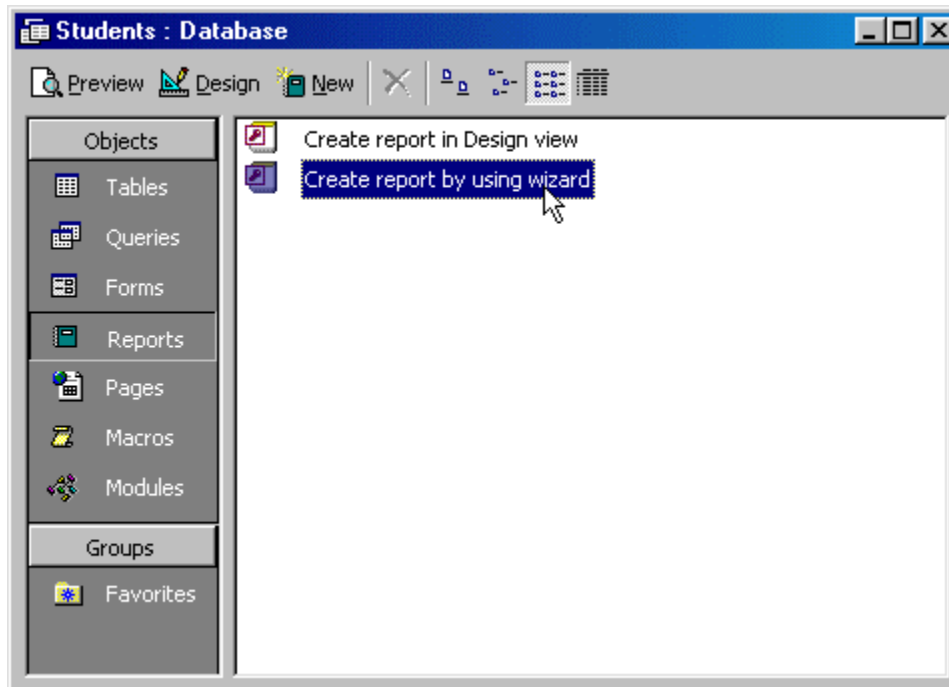
Once you have saved a query, you can run it at any time from the main database window by clicking the **Open** button after highlighting the name of the query you wish to run:



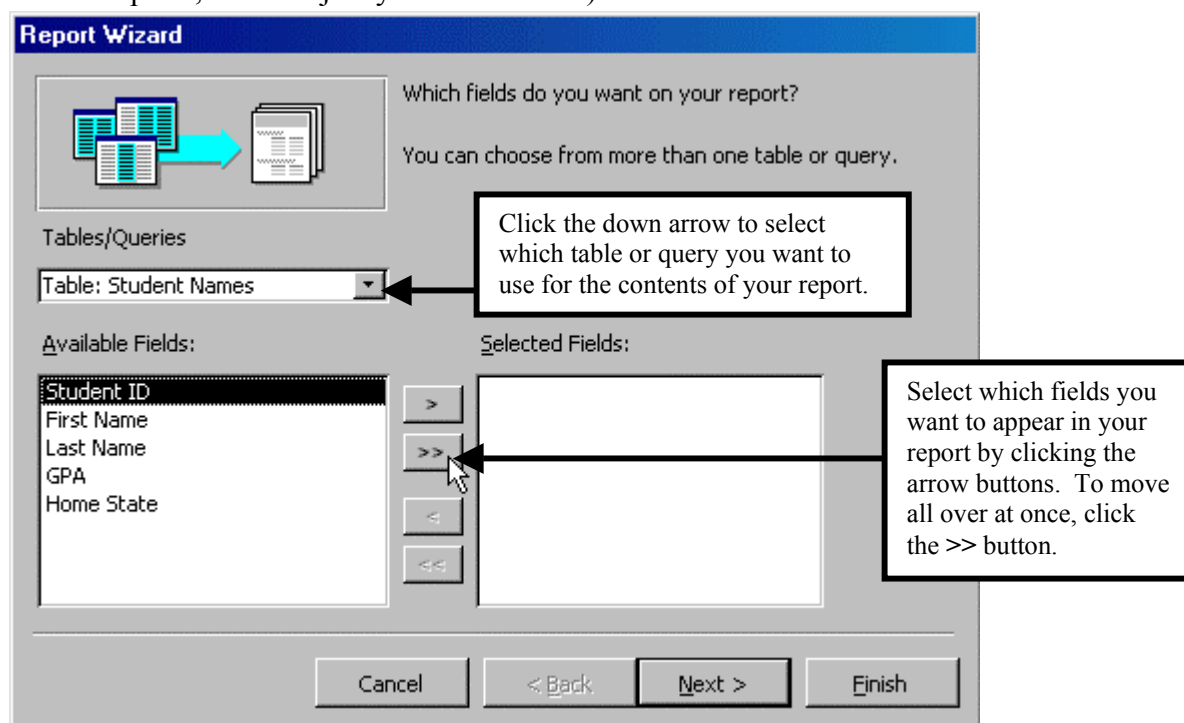
REPORTS

Reports are the means by which you produce nicely formatted and labeled listings of your database information. Unlike working from the Table Datasheet View, a report allows for much better formatting, improved data manipulation, and a cleaner look overall.

To create a report in *Access*, click on the **Reports** button in the main database window, then double-click **Create report by using wizard**, as shown on the next page:

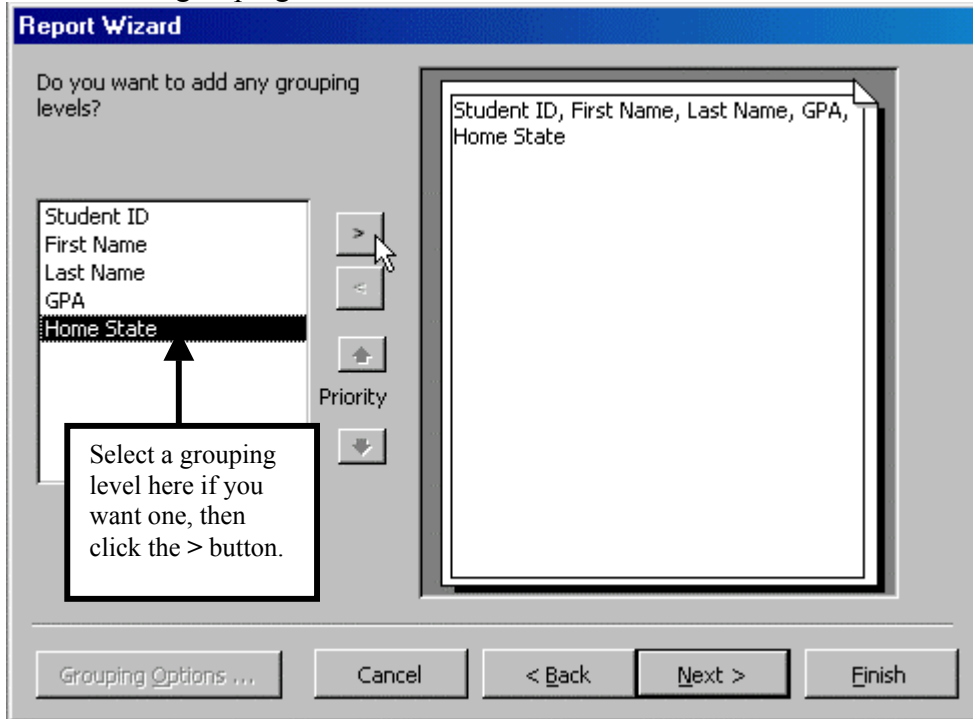


From the following **Report Wizard** dialog box, click the down arrow under **Tables/Queries** to select the table or the query you want to base your report upon. (Yes, query results can be the basis of reports, too--not just your data tables.):

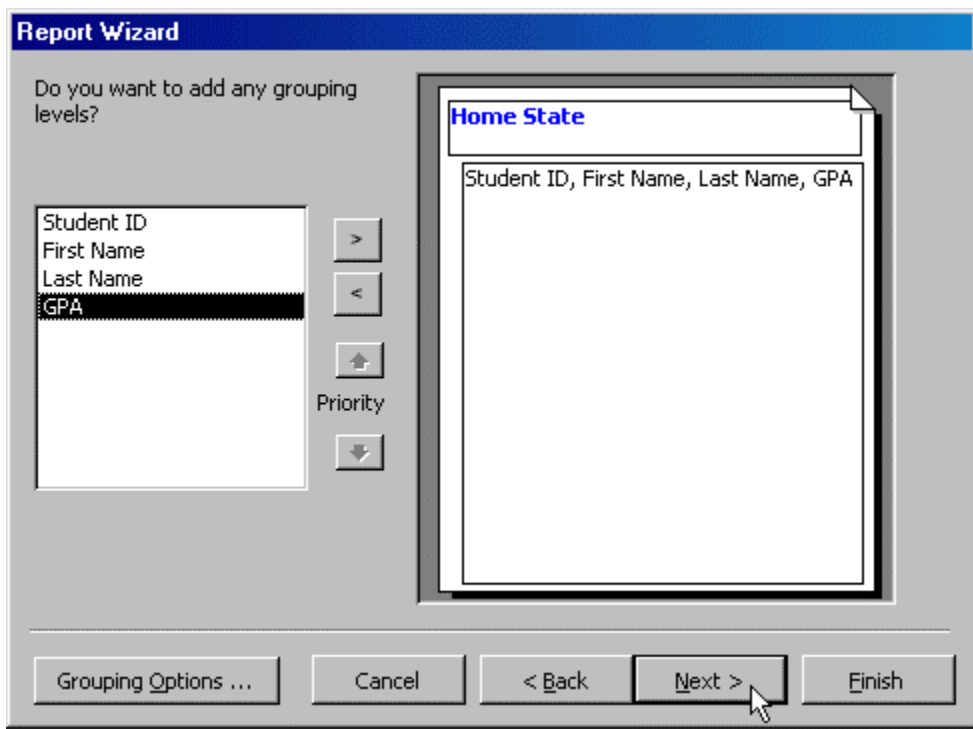


Note that this dialog box looks very similar to the Forms Wizard we saw earlier. You are to select which field(s) you want to appear in your report by selecting the field names on the left side of the window and moving them over to the right. You can click the > button to move the fields over one at a time, or the >> button to move them all over.

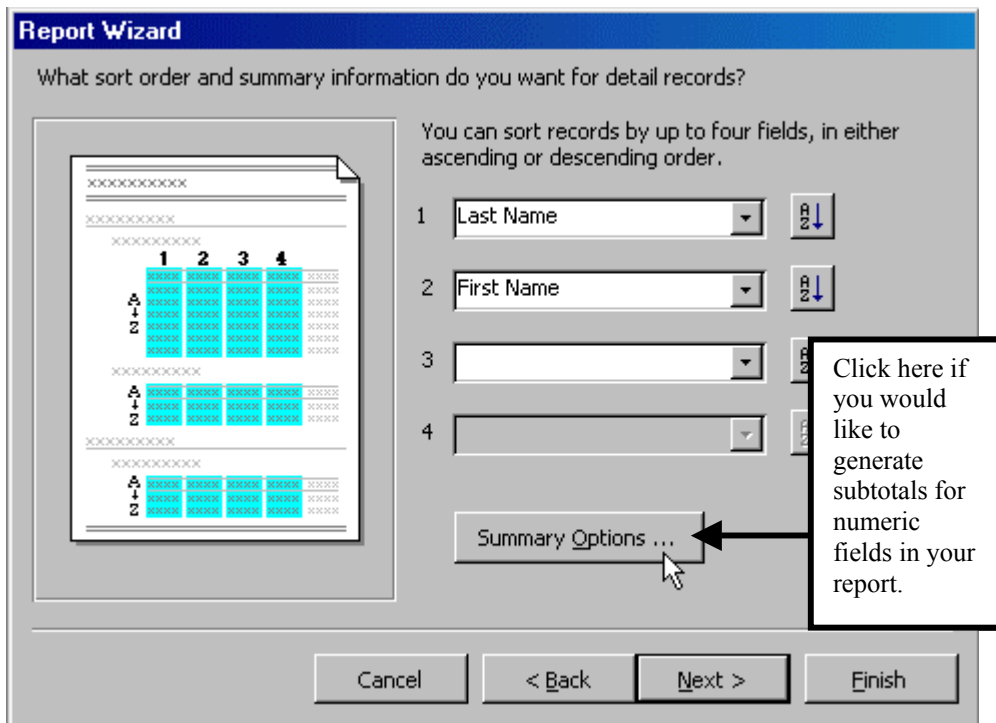
After you have selected the fields, click the **Next** button, and you'll be asked if you want to add any Grouping Levels. This is not necessary, but it *does* enable subtotals to be generated in the report for each group. For example, if you wanted the average GPA by state of residence, you should add a grouping level of state:



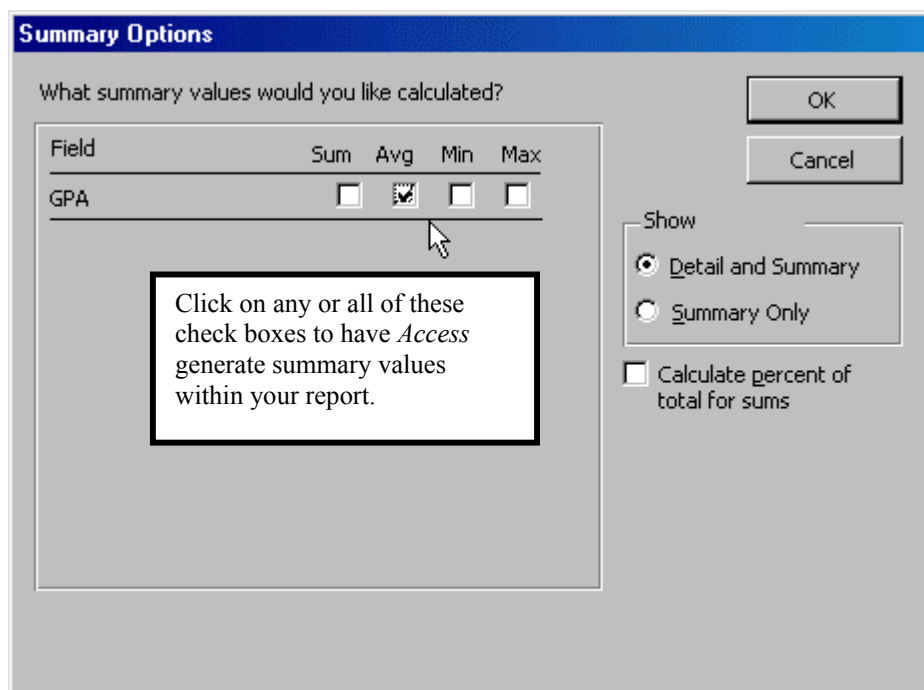
Once selected, the dialog box will look like this:



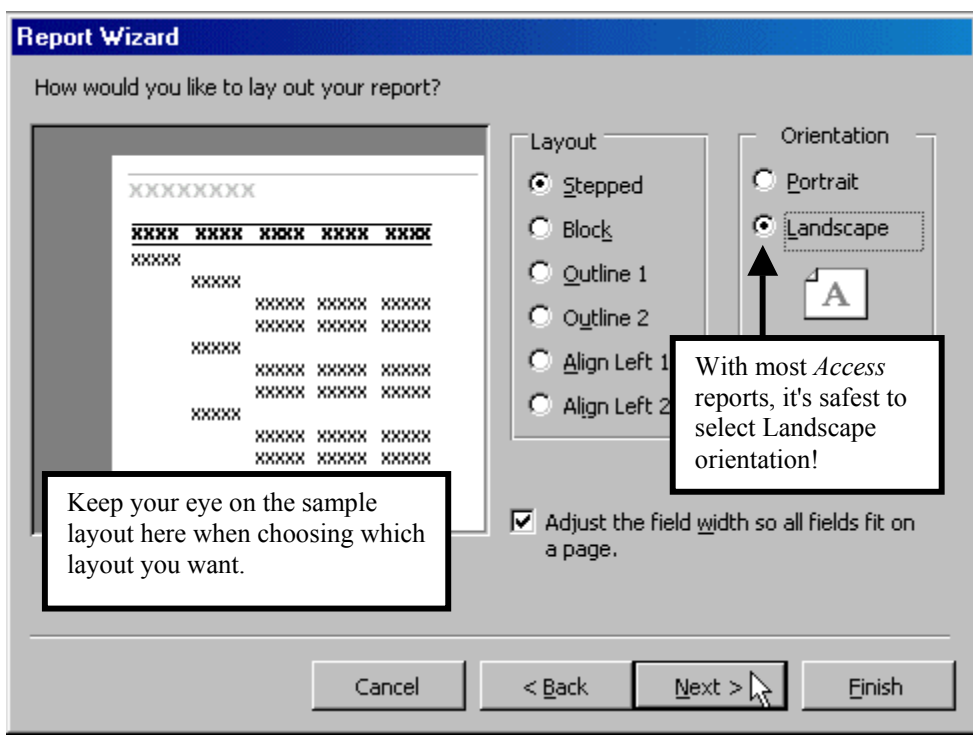
Click the **Next** button, and you'll see a screen prompting you for sorting information. Here, we are specifying that within each state, we want names to appear in alphabetical (A-Z) order by last name, and then within any last names that match in alphabetical order by first name:



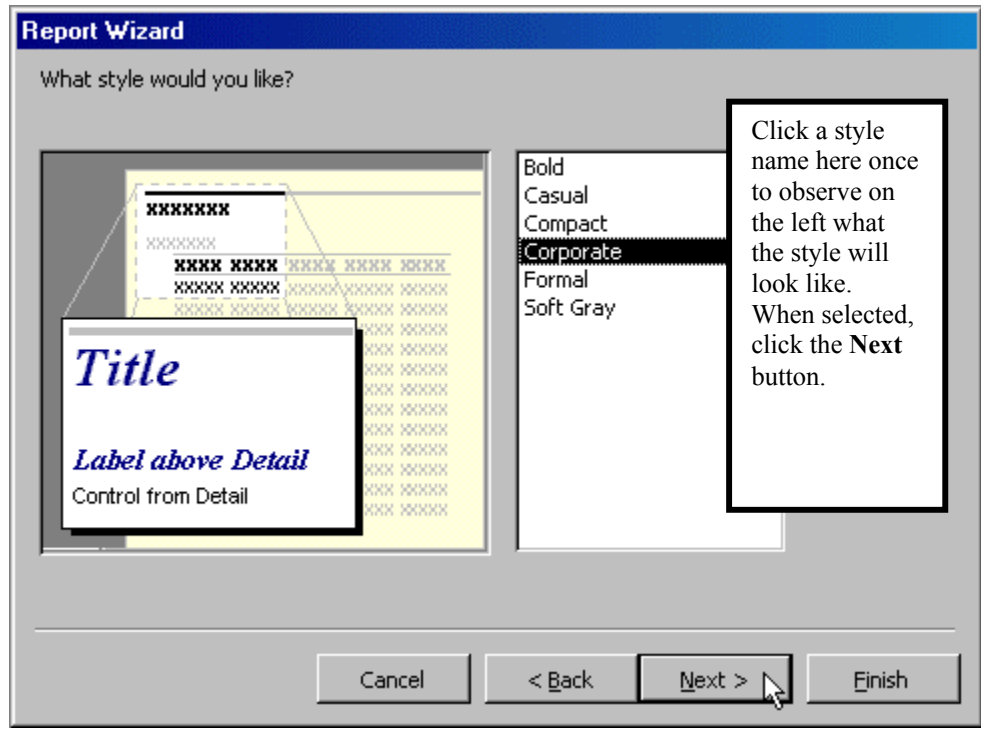
Notice that the above dialog box has a button labeled **Summary Options**. This is available *only* if you decided to add a grouping level earlier, and is good for generating various subtotals within your report. If you choose to click **Summary Options**, here is what you will see:



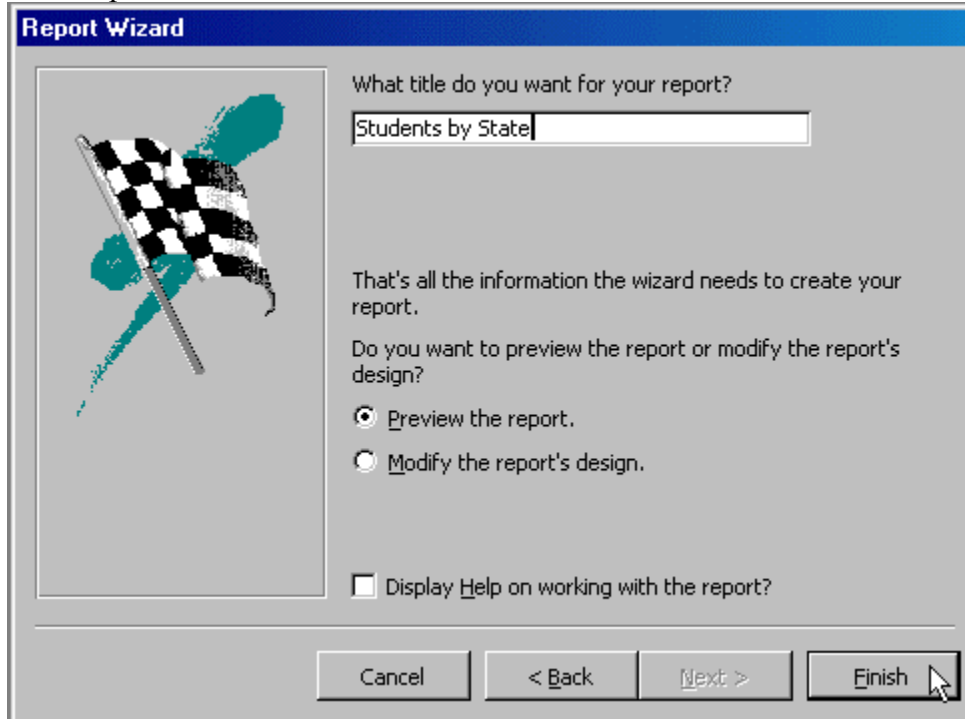
Since we want the average GPA for each state to be generated, we've clicked in the **Average** box. After clicking **OK**, we return to the sorting order box we saw previously, and can click the **Next** button to go on, when we are asked about our layout:



Generally, you want to select **Landscape** orientation since *Access* reports often contain many fields. Even then, you will not always fit everything onto one page. After selecting the printing orientation, you have one last appearance question to answer:



After selecting the general appearance of the report, click the **Next** button, and you are at the final step:



You can type whatever you want to as the Report name. This is how it will appear from now on under the **Reports** button of the main database window. (One popular naming convention is to type **rpt** before a report name, and to eliminate spacing, such as: **rptStudentNames**. However, since the name you type here is what appears at the top of the report, you will then likely want to go into the report **Design View** and change the report header manually.)

Finally, click **Finish**, and you will see your report in the Preview mode, as shown on the following page:

Microsoft Access - [Students by State]

File Edit View Tools Window Help


100% Close W

Students by State

<i>Home State</i>	<i>Last Name</i>	<i>First Name</i>	<i>Student ID</i>	<i>GPA</i>
<i>IL</i>				
	Adams	Samuel	12345	3.45
	Bare	Teddy	98765	2.98
	Cheatum	Dewey	98989	2.01
	Dahl	Barbie	33224	3.23
<i>Summary for 'Home State' = IL (4 detail records)</i>				
Avg				2.92
<i>IN</i>				
	Rabbit	Jessica	33333	3.50
	Rabbit	Roger	22222	3.72
	Springseternal	Hope	32154	1.75
<i>Summary for 'Home State' = IN (3 detail records)</i>				
Avg				2.99
<i>WI</i>				
	Badenov	Boris	13131	0.03
	Doe	John	44332	2.99


Click here to go to Design View.

Note how--as we specified when constructing the report with the Report Wizard--that our students are grouped by state, and within each state they are in alphabetical order by last name, and then by first name. Also note that the average GPA appears calculated for each state, as we specified in the Summary Options section.

You can make changes to virtually everything on this report by clicking the **View** () button on the toolbar, which brings you to the **Design View**. There, you can click (for example) in the **Report Header** section to change the title of the report and make other changes to the font type, size, etc. In addition, in the Design View, you can click the **View** pull-down menu, then **Toolbox** to insert additional fields, labels, etc. You can also create other calculated fields within your report by a creating text box and changing its *properties*. It is beyond the scope of this introduction to give additional details, but most books on *Access 2000* will give you help on this.

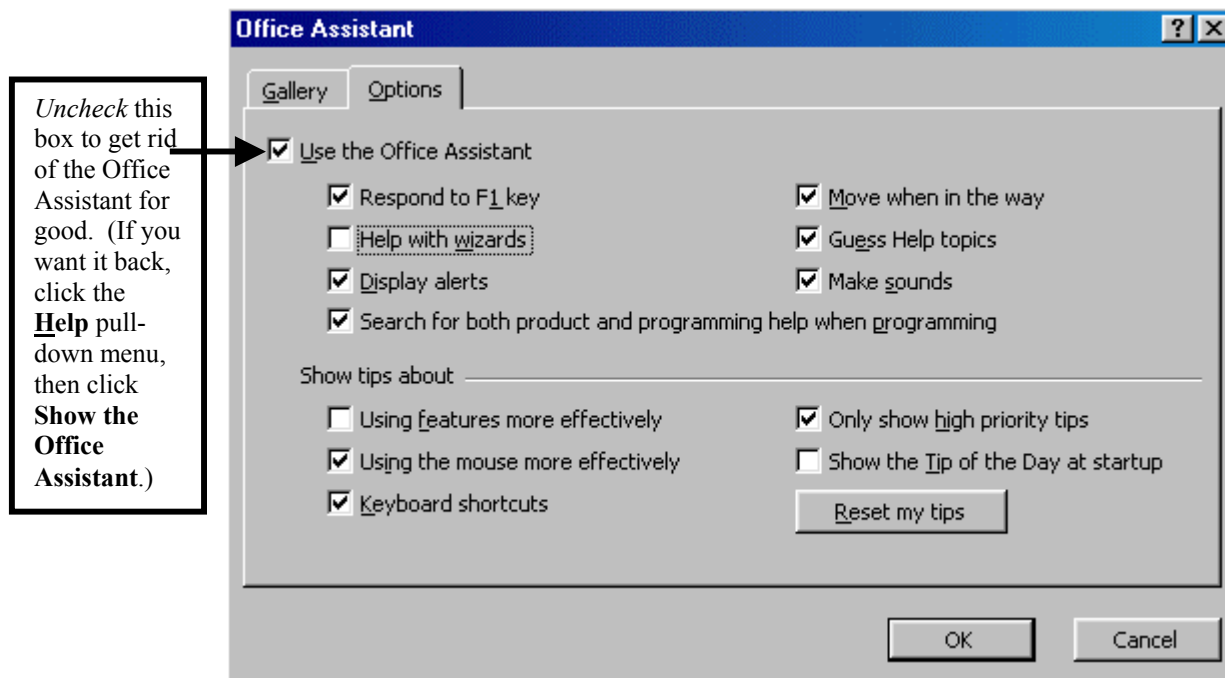
For additional information on creating calculated fields in *Access 2000*, consult our separate handout, titled *Creating Calculated Fields in Access Queries and Reports*.

HELP FACILITY

Like every other *Office97* program, *Access* has a variety of ways to find help. The most direct is clicking on the **Help** button on the toolbar: 

This brings up the Office Assistant, where you can type keywords of questions into a box. You can also click the **Help** pull-down menu, and click **Microsoft Access Help**. To get rid of the Office Assistant, *right-click* over it, then left-click **Hide**.

For more conventional help (without the Assistant), when the Office Assistant is showing click the **Options** button in the Assistant's help window, which will lead to this screen:



This document was originally conceived by Margaret Kozak of Dominican University's Computer & Information Sciences Department, and updated by Ken Black, Director of Teaching and Learning Technology. June 2001.