FIT 100
More Microsoft Access and Relational Databases
Creating Forms for Data Entry

Table of Contents

Reading for Lab:
- Review Chapter 13 of *Fluency with Information Technology: Foundations and Fundamentals of Information Technology*
- Check out the Database Definitions on the Web Reference page: http://courses.washington.edu/fit100/au02/webref.shtml

Introduction:
In today's lab we will look at the presentation layer: creating forms for users to input data and receive data output. (See the first database lab for a brief description of all three levels involved in a complete Relational Database Management System)

Presentation Services
The interfaces created to allow users of the Database system the ability to enter data and to view output of data. A *Form* is the typical interface used for input and output of data that is retrieved by a Query. There are other interfaces in Access such as Reports and Data Access Pages, but you will only be dealing with Forms for the first part of Project 3.

Objectives:
1. Understand the separation of data as it is stored (in tables) and as it is displayed for users for update or to disseminate.
2. To create forms that will allow a user to add data to tables using a friendly interface.

TO BEGIN:
Download a copy of your **YourName.mdb** database to the desktop from Dante and open it.

If your database was not closed when you tried to ftp it after the last lab, you will have an empty file. If this happened, you will need to work through the first database lab to recreate it.

Working With Forms in Access

Today's lab allows you to work with a familiar object, the Form. In Access forms are used for presentation in a manner similar to their use in JavaScript. Both are intended as a way to present data of some kind to a user or obtain data from a user in a way that is intuitive and helpful (although not every form is successful in this endeavor!).

The first form will be created using a wizard. The second and third will be created just in design view. In addition, you will add combo boxes on one form in order to view data from two different tables as we add data to a third. You will also:

- Create a list box to navigate to individual records based on a single attribute.
- Add buttons to refresh and close the form
- Add combo boxes to view data from one table in order to help the user add data to another

1. Create a Form using the Form Wizard

A. Create a form to add Major instances (rows of data) to the Major table. Go to the Forms window and create a form for the Major table using the form wizard.

B. Select **Table: tblMajor** from the available fields and move all the attributes over to the right to the selected fields.

C. Choose a layout (Usually Columnar or Justified is the nicest looking)
   - Click Next
   - Choose a style
   - Click Next

D. In the last window, give a name to the form. Call it **frmMajor** to distinguish it from the table Major. Click finish and the form will open up, ready for you to add data.

E. Using **frmMajor**, add the following Majors:

MajorName	School_Dept
INFO	The Information School
CSE	College of Engineering
BIOL	The Biology Program
CHEM	Chemistry Department
PHYS	Physics Department
SPAN	Romance Languages and Literature
ART	Art Department
HIST	History Department

F. Save the form layout and close the form. Go to the Major table. Are the records you just added through the form in the table? Good!
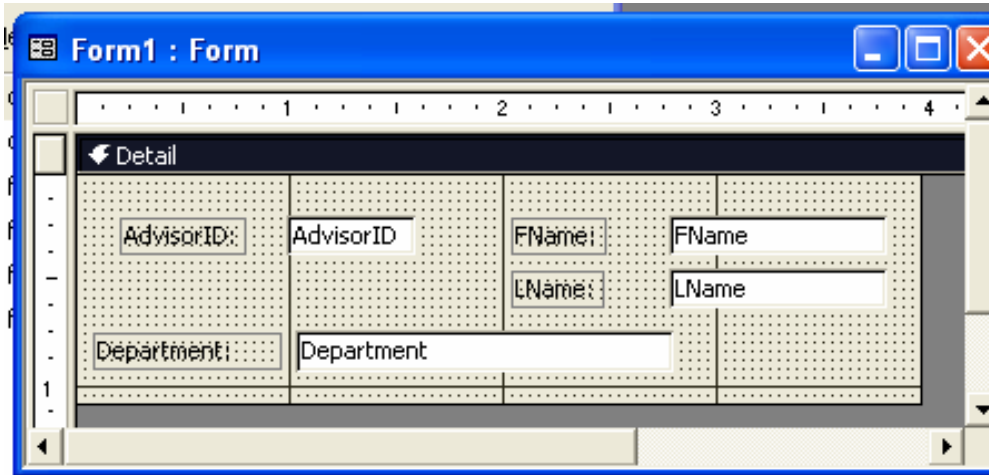
2. Creating a Form Using Design View

A. Create a form in design view to add data to the Advisor table. This allows you to place all objects on the form and format it by hand. Name it **frmAdvisor.**

B. To add the fields to the form, simply access the Form properties
- Right click on the form title bar, select Properties
- Make sure the Form object is selected in the window
- Select as the record source the tblAdvisor table from the list.

Form Title Bar
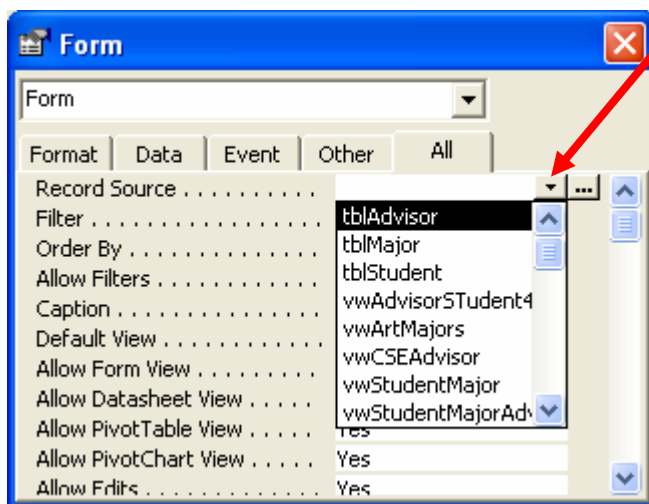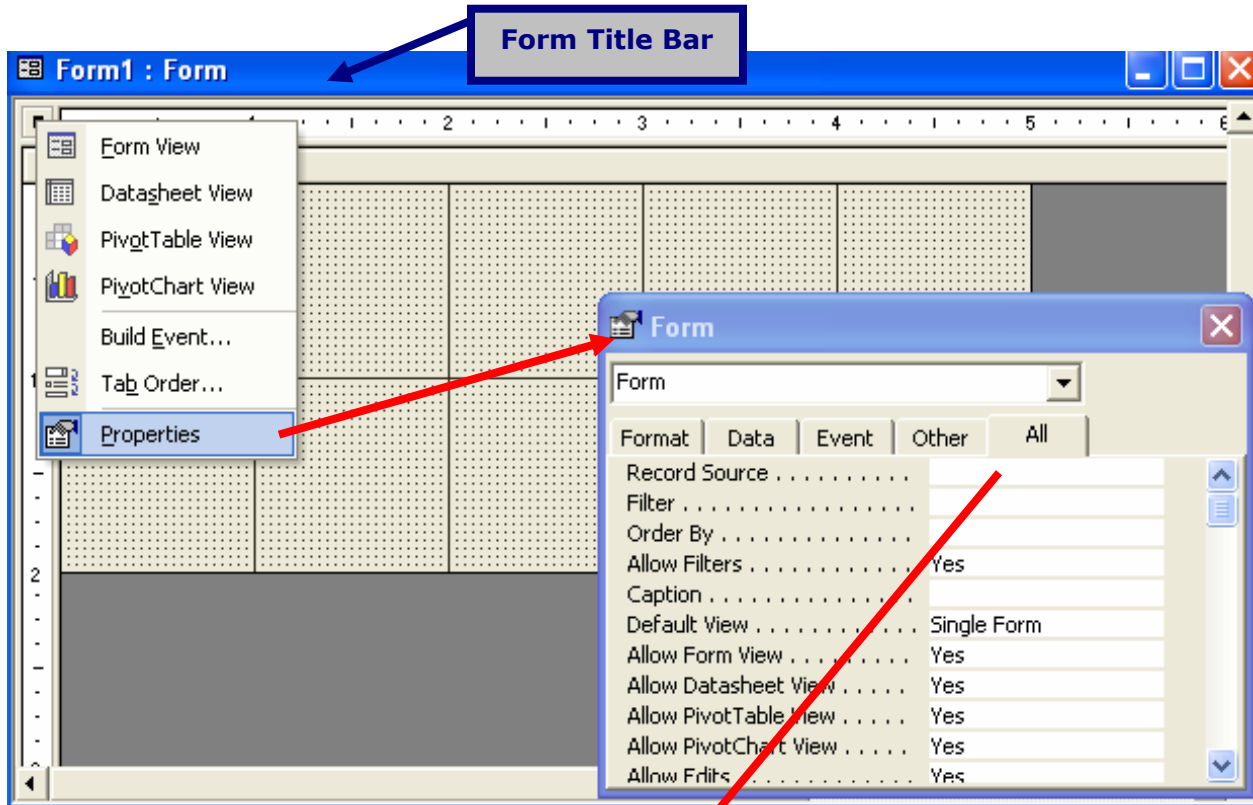
C. A window with a list of the attributes for the Advisor table should pop up. If it doesn't:
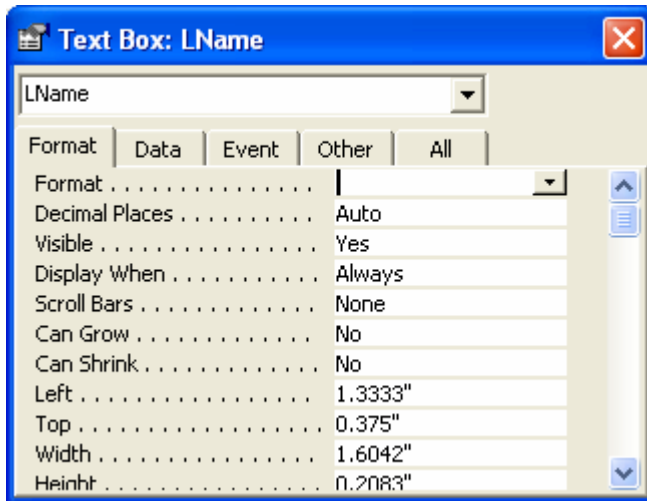- Go to **View>Fields List** on the Menu bar

tblAdvisor
AdvisorID
LName
FName
Department

D. Drag and drop the attributes onto the form in an arrangement that suits you.

Form1 : Form

Detail

AdvisorID: [AdvisorID]   FName: [FName]

LName: [LName]

Department: [Department]
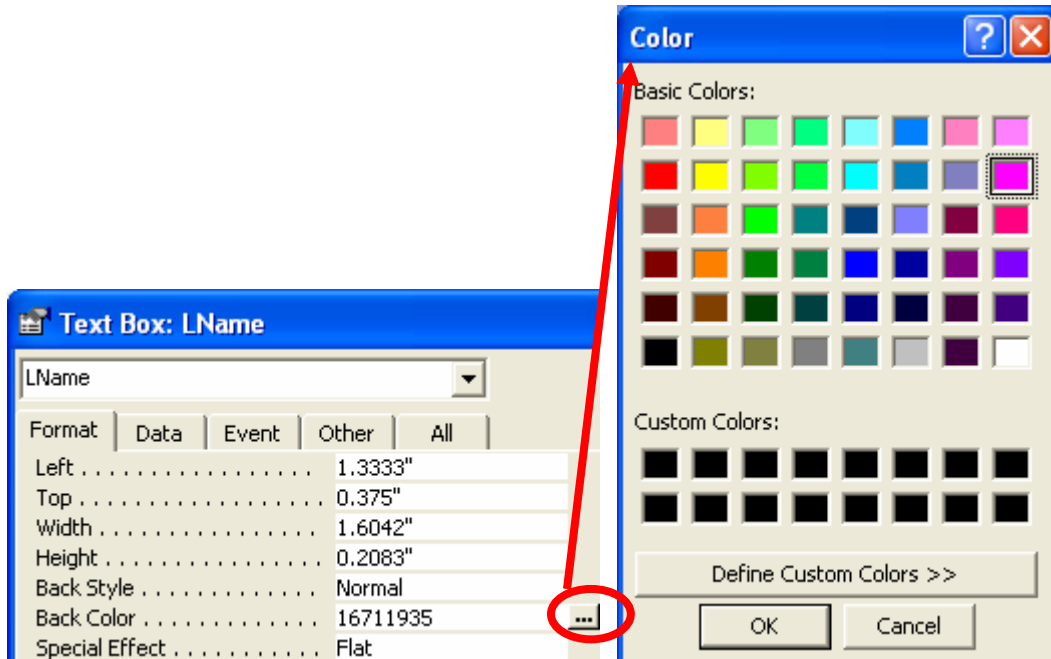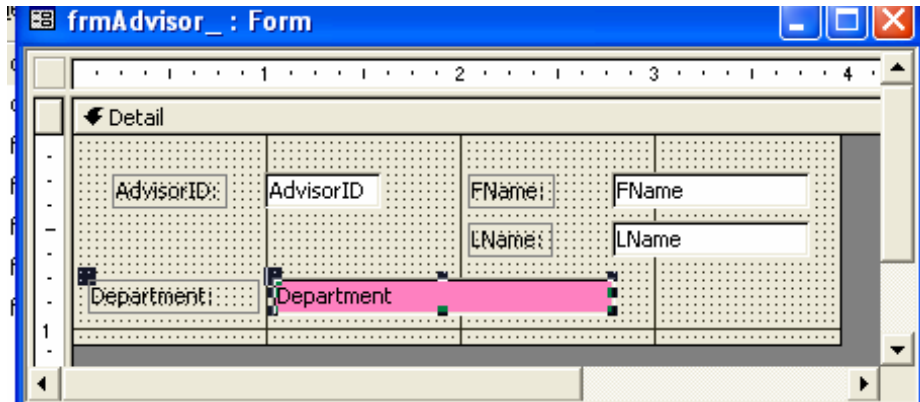
E. Switch to Form View and see the how it looks.

3. Manipulate field properties in Design View

A. Switch back to the design view and select one of the text box fields shown in the form.

B. Right click and go to the properties of that field.

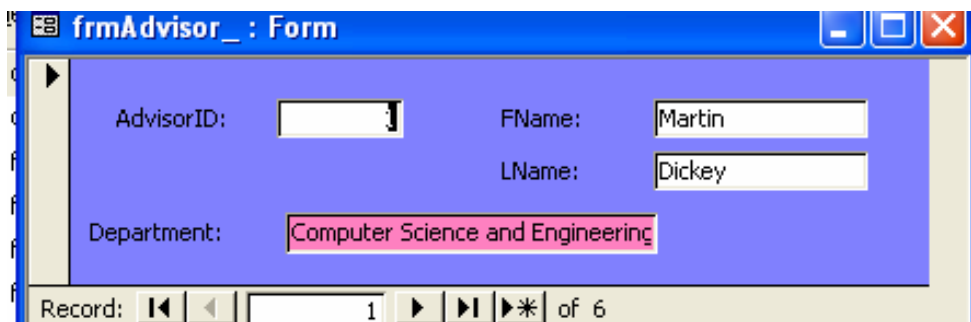**Text Box: LName**

LName

| Format | Data | Event | Other | All |

Format . . . .
Decimal Places . . . . Auto
Visible . . . . Yes
Display When . . . . Always
Scroll Bars . . . . None
Can Grow . . . . No
Can Shrink . . . . No
Left . . . . 1.3333"
Top . . . . 0.375"
Width . . . . 1.6042"
Height . . . . 0.2083"

C. Explore the properties and change a format based property, such as Caption or BackColor.

Color

Basic Colors:

Custom Colors:

Define Custom Colors >>

OK   Cancel

Text Box: LName

LName

| Format | Data | Event | Other | All |

Left . . . . 1.3333"
Top . . . . 0.375"
Width . . . . 1.6042"
Height . . . . 0.2083"
Back Style . . . . Normal
Back Color 16711935
Special Effect . . . . Flat

D. Now adjust the back color of the Form, then look at it in Form View

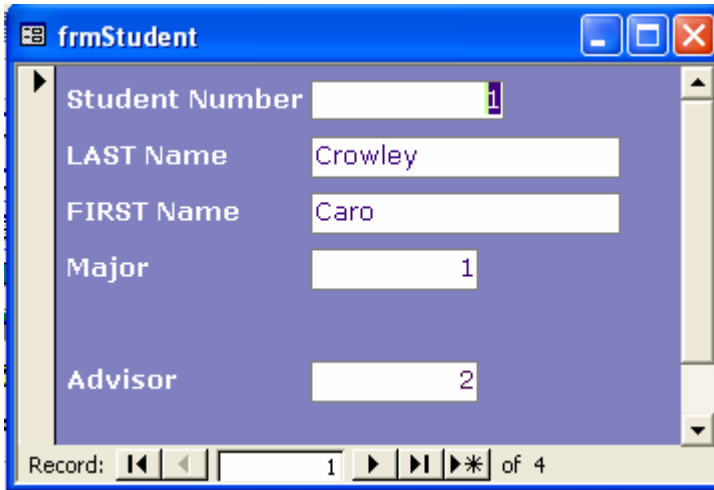E. Using frmAdvisor, add the following Advisor Names and Department:

Mel Oyler from Informatics
Alan Borning from Engineering
John Cusak from Art
Vitoli Demetrio from Modern History
Garbonzo Bean from Physics

Go to the Advisor table. Look at the records you just added through the data entry form.

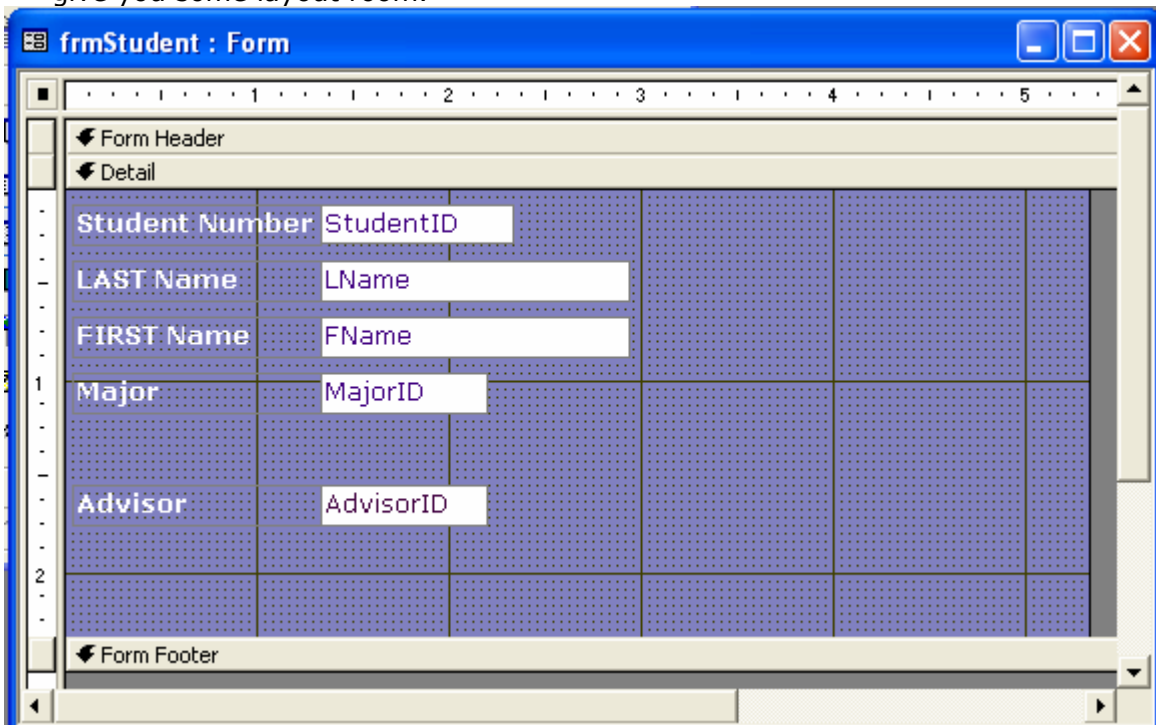## 4. Displaying Data from One Table to Help Add Data to Another: Adding Instances to tblStudent

Now we'll do something a little different: create a form that adds data to the Student table, but it will *display* data from Major and Advisor tables.

A. Go through step 3 again, but use the **Table: tblStudent** to add all the fields.

B. Save the form as **frmStudent**. You now will have a form that may look different from this, but all the attributes should be listed:

C. Switch to the design view by clicking the icon in the upper left corner. In design view, extend the borders of the form and the form footer to give you some layout room.

D. Add two combo boxes to the form. One combo box will be used to display the contents of the Major table. The other will display the contents of the Advisor table.

E. From the tool box on your screen, select the combo box object.
   • Add the first combo box to the form by clicking once on the tool.
   • Move to the area of the form where you would like to have the combo box and use the "click and drag" action with your mouse to draw the combo box to the size you want on the form.

Once you place the combo box control on your form, a combo box wizard window will come up to walk you through the process.

Select the following options from each screen and then click Next:

1. Have the combo box look up values from a table
2. Select the Major table for the table that will provide values for the combo box.
3. Move all available fields over to the selected fields window.
4. Make sure the "Hide key column" box is checked
5. Store the value in the "MajorID" field of the Student table (those are the options in the drop down menu)
6. Leave the name the combo box is given and click Finish.

You will return to design view. There is now a combo box and label in the form. The label can be deleted (the combo box should line up across from the other Major label and text box).

F. Switch to Form view and look at the values that are in the combo box.
   Tres cool, no?

G. Return to Design View. Add another combo box just across from the AdvisorID text box.

H. Select the following options from each screen and then click Next:

- Have the combo box look up values from a table
- Select the tblAdvisor as the table that will provide values for the combo box.
- Move ONLY the **AdvisorID**, **Lname** and **Department** from the available fields over to the selected fields window.
- Make sure the "Hide key column" box is checked
- Store the value in the "AdvisorID" field of the Student table (those are the options in the drop down menu)
- Ignore the name option for the combo box and click Finish.

I. Return to design view. There is now a combo box and label in the form. Delete the *label* not the text box! (Ask your TA if you aren't sure)

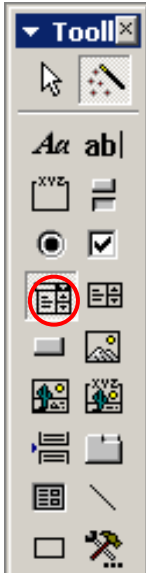J. Start moving through the records you have already created in Form View.

You should see the number value of the Advisor in the text box for Advisor ID and the name and department of the person that goes with that number appears in the combo box on the right.

We have set up the combo box so that it displays the text values for us to see, but it will store the corresponding key value in the Student table in the attribute AdvisorID. That stored value will be referred to as a foreign key in the Student table.

K. Since we deleted the original Major attribute and that held text earlier in the lab, let's now add a major for the students in the table. Use the combo box to select a Major. Notice the corresponding key value is stored to the left.

L. Add a total of 6 student records using the form. Give the students a major and an advisor.

5. Add a List Box to the Form

A. Add a list box to the form. This list box will be used to access a particular record using only the last name field

B. From the tool box on your screen, select the list box object (also known as a **control**). Add the list box to the form here by clicking and dragging it to the size you want on the screen.
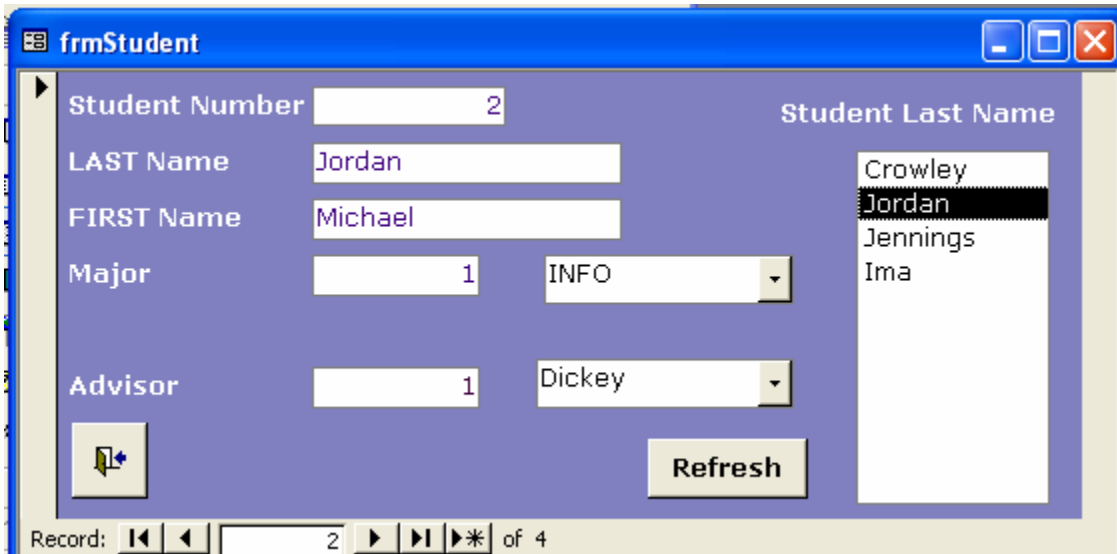

Once you place the list box control on your form, a wizard window will come up to walk you through the process.

Select the following options from each screen and then click Next:

7. Have the list box find a record based on the value selected.
8. Click Next
9. Select the field to use from the student table [LName].
10. Move the field over to the selected fields window.
11. Click Next.
12. By Default, the Key field is also added if you did not choose it. Make sure the "Hide key column" box is checked.
13. Name your list caption "Student Last Name".
14. Click Finish.

You will return to design view. There is now a list box and label in the form.

C. Switch to Form view and look at the values that are in the combo box. Tres cool, no?

D. Start moving through the records you have already created by selecting a name from the list box.

6. Refreshing the Form and Closing the Form

A. Go to design view and add two command buttons to the form. One will refresh the page and the other will close it when a user is done adding data.

B. Select a command button control and place it on the form in the upper right hand corner.

C. When the Command Button Wizard comes up, select **Form Operations** from the Categories column and **Refresh From Data** from the Actions column and click next:
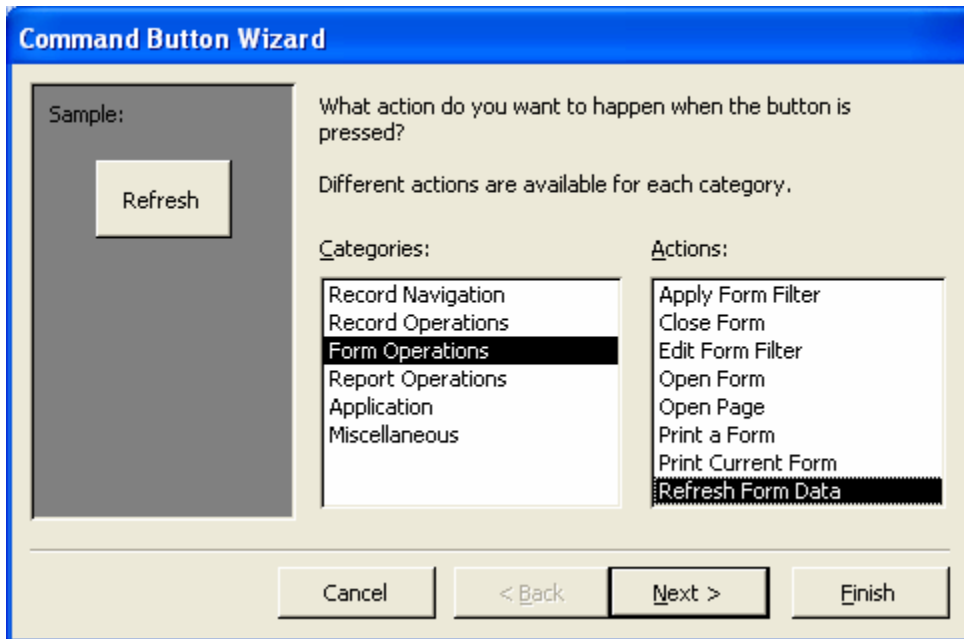
D. Choose either a picture or text for display on the button and click next:
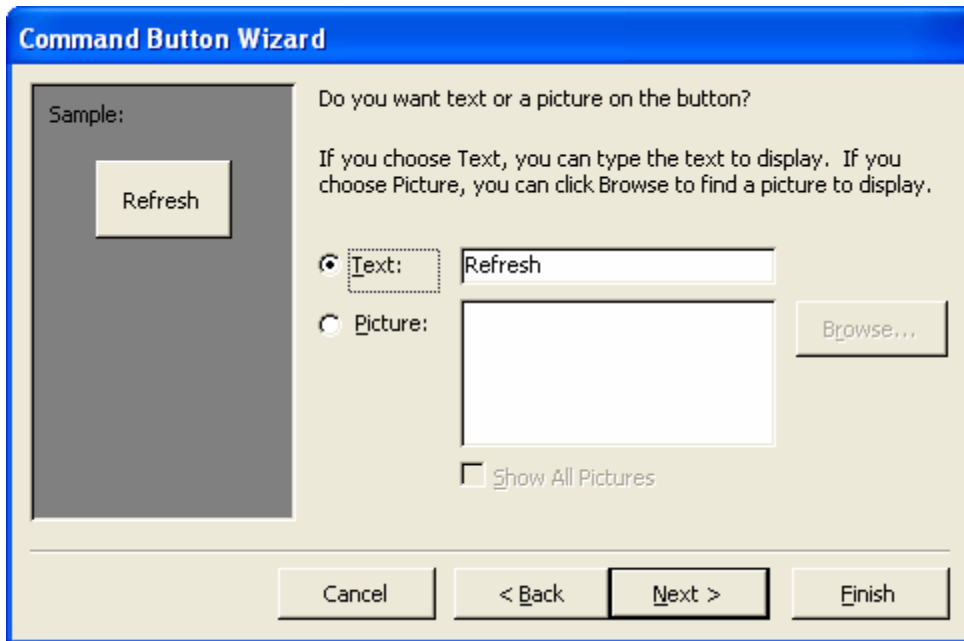
E. Name the button **cmdRefresh** and Finish.

F. Save your form. Switch to Form View, add an additional record and press the Refresh button.

- Does the list box get updated?

- Are the names in the list box updated in alphabetical order?
  No, they are updated and displayed in the order they were entered to the table. We'll look at ways to sort data alphabetically next lab.

G. Now go through the same process but select the correct operation to close the form when the button is clicked.

H. Save your form. Switch to Form View and press the exit button. Does the form close?

A quick note about file size

As you start working on your project, you will create forms. You will make mistakes and decide to delete forms and other objects from the database.

In other applications, this would usually mean your file gets a little smaller. It is NOT that way in Access.

Access "remembers" all the different forms and command buttons, etc that you create - even after you delete them. If you don't occasionally do a little "house cleaning", your database file will grow immense!

To clean up your database and decrease the file size after you have been working in it for a while, compact and repair the database:

Tools>Database Utilities>Compact and Repair Database...

Do this to keep the database size manageable. ESubmit will only accept files up to 2 MB in size, so you must keep your database under control for Project 3.
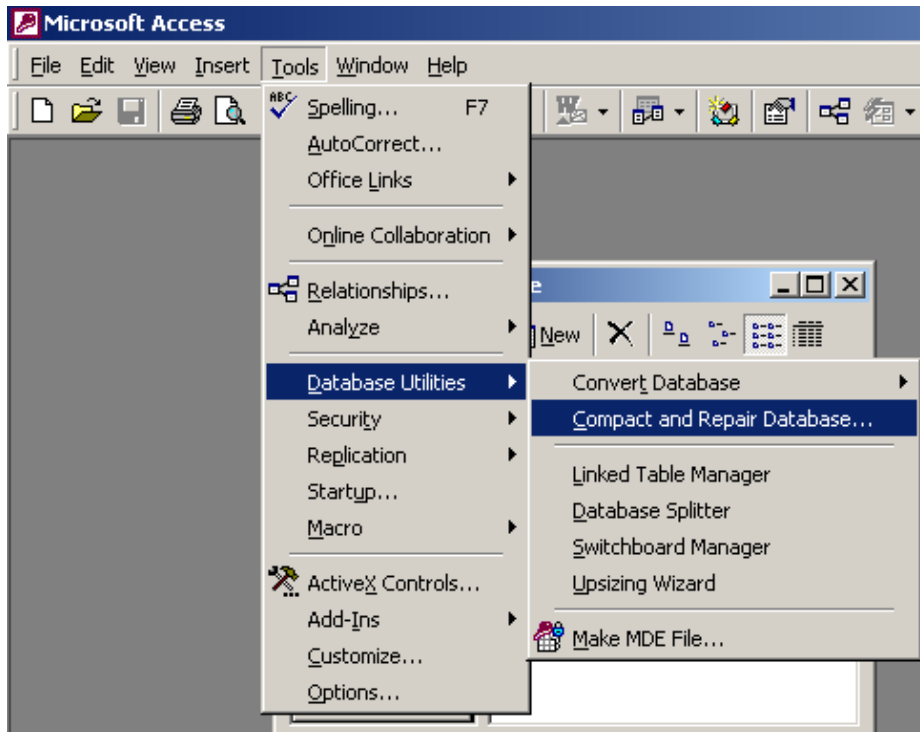
Close your database and FTP it to your Dante account if you don't already have it on disk. We will use it again for the next lab.

WHEW!!!! You've made it through another lab! Only a few more to go!!!!!

7. Lab Questions (use to test your understanding, not turned in)

A. In Relational Databases, a table holds rows of data about one particular entity, like student. Each row is uniquely identified by a key. In this lab we are using a tool in access called autonumber to generate a unique key for us. If we weren't using the autonumber feature and we didn't have student IDs for each student, what other attribute(s) could be used as the key so that each row in our student table is unique? (Hint: You probably have to use a combination of attributes).

B. Explain how your chosen key is ensures the row is unique.

C. What is the form used for in this lab? In other words, what is the form doing as we enter data into it?

D. How are we able to use the combo boxes on the Student form and have the database system "know" that a major name displayed in the combo box matches the number showing for MajorID on the Student Form?

14