

## Working with Forms

### *Aims.*

You should be able to:

Design forms to Add, Edit, Delete and View data on screen

Base forms on both tables and queries

Design and create a simple user interface to integrate the different aspects of the student records database.

### *What is a Form?*

A form is an object that allows us to:

- more effectively carry out data entry
- create custom dialogue boxes to accept user input ( and to act on that input)
- Create a menu system to access other forms and/or reports

A form gets its information from the records in the underlying tables (either directly or via queries). However the presentation on screen can be tailored to the level of expertise of the user. This means that a skilled database designer or programmer can make an inherently complex construction (the database) appear simple to the user.

### **Forms can contain:**

Graphics objects: (lines, boxes, pictures etc)

Text boxes: (these usually get their data from underlying data fields - i.e. they are BOUND to a data field).

Text labels: (used to make the layout of the form clearer to the user)

Calculated fields: (Used to display calculated values that are not stored directly in a data table)



Any form will have at least one link created between it and some underlying record source. This linking is done by using graphical objects called controls such as text boxes, labels, and command buttons (mentioned on previous page). There are more controls, refer to the Microsoft Access documentation for details.

There are three types of control, BOUND, UNBOUND and CALCULATED.

A BOUND control gets its value from a data field in an underlying table.

An UNBOUND control is not linked to an underlying data field and can be used simply to display instructions to the user or to accept user input.





A CALCULATED control displays the result of an expression. This is recalculated each time the form is displayed or a relevant value changes.

## *Creating a Simple Form*

We are going to create a form that allows an on screen display of the contents of the STUDREC data table. We will be able to add, edit and delete data as before.

The point of creating this form is to do with the way in which users interact with the database that we are designing for them. It's largely a question of producing an ergonomically effective interface.

Work through the following example to create the form.

-  Ensure that the database window is open and that you have selected the FORMS tab.
-  Press the NEW button
-  Select the "Form Wizard" and base the form on the STUDREC table
-  Press the OK button to continue with the Form creation





Select all of the data fields

Note: that it is possible to select only some of the data fields in STUDREC. If we wished to omit some of the fields, it is possible that we would not be able to update the database. In order for the database to be successfully updated we would need to be careful to include any key field(s) and any required fields in our form.



Press the NEXT button

We are now asked to define the Form layout, the choice is between Columnar, Tabular, Datasheet, and Justified. (See Figure 4.1, Figure 4.2, Figure 4.3 and Figure 4.4 for examples of these layouts.



Select Columnar



Press the NEXT button

We can now select a colour scheme for our Form.



Select STANDARD



Press the NEXT button



Call the Form "All Students (Personal Details)" - don't type the quotation marks.



Press the FINISH button to display your form

Your form should look similar to that shown in Figure 4.1.





Notes

tabular layout									
SREF	INT	SNAME	DOB	GENDER	FES	KIDS	HTOWN	Distance	
1	TJ	OSMAN	29-Sep-53	M	No	0	WILHOUSE GF	9	
2	S	LANGLEY	21-Aug-57	F	No	0	HUDDERSFIELD	11	
3	H	WILSON	07-Jul-62	M	Yes	1	HUDDERSFIELD	14	
4	I	LALILI	21-Mar-54	M	Yes	2	HALETHORPE	11	
5	A	JONES	10-Nov-43	F	Yes	2	SHEFFELD	14	
6	S	ISHMO	15-Dec-51	M	No	0	LEEDS	31	
7	K	ARNOTT	01-Aug-61	F	Yes	2	SHEFFELD	14	
8	II	ANNI I	21-May-61	M	Yes	1	LEEDS	14	
9	N	GFEEN	30-Sep-53	M	Yes	1	SHEFFELD	14	
10	H	JACKSON	27-Apr-41	M	Yes	0	SHEFFELD	11	
11	A	ARNOTT	23-Aug-54	M	Yes	2	BARNSELEY	3	
12	N	IIY	11-Nov-57	M	No	11	BRISTOL	1	
13	K	WILSON	13-Mar-61	M	No	2	BARNSELEY	3	
14	J	BROWN	29-Sep-53	F	Yes	1	BARNSELEY	3	
15	A	ARNOTT	23-Aug-54	F	No	1	BARNSELEY	3	
16	G	WILLI	11-Mar-61	M	Yes	11	HALETHORPE	11	
17	J	GFEEN	06-Aug-63	F	No	0	BARNSELEY	3	
18	J	GFEEN	09-Aug-43	M	No	0	BARNSELEY	3	
19	F	WATECH	03-Mar-53	M	Yes	1	THURLESTONE	7	
20	I	LALIM Y	11-Aug-54	M	No	2	THURLESTONE	7	

Record 1 of 41

Figure 4.2: Tabular Form Layout



Notes

☰ All students (personal details) - datasheet layout

SRE	INIT	SNAME	DOB	GENDER	RES	KIDS	HTOWN	Distance
1	U	DEMAN	29-Sep-11	M	No	0	MILITARY_GHLLA	9
2	S	LANGLEY	21-Aug-67	F	No	0	HIDDERSFIELD	14
3	H	MILSON	07-Jul-62	M	Yes	1	HIDDERSFIELD	14
4	.	CARTER	21-Mar-64	F	Yes	2	BARNSELY	3
5	A	JONILE	16-Nov-40	F	Yes	2	SILFHILL	14
6	S	SHEMC	05-Dec-60	M	No	0	LEEDS	34
7	K	ARNOTT	07-Aug-60	F	Yes	2	SILFHILL	14
8	B	ARNOTT	28-May-62	F	Yes	1	LEEDS	34
9	M	GREEN	10-Sep-60	M	Yes	1	SILFHILL	14
10	H	JACKSON	21-Apr-41	M	Yes	0	SHEFFIELD	14
11	A	ARNOTT	21-Aug-64	M	Yes	2	DARBUSLY	J
12	N	HEY	10-Oct-65	F	No	0	SLKSTONE	1
13	K	MILSON	11-Mar-65	M	No	2	DARBUSLY	J
14	.	BROWN	29-Sep-63	F	Yes	1	BARNSELY	3
15	A	ARNOTT	21-Aug-64	F	No	1	DARBUSLY	J
16	G	WHITE	03-Mar-65	M	Yes	3	BARNSELY	3
17	.	GREEN	06-Aug-69	F	No	0	DARBUSLY	J
18	.	GREEN	09-Aug-45	M	No	0	BARNSELY	3
19	I	WATSON	01-Mar-60	M	Yes	1	THURLSTONE	7
20	L	HARVEY	08-Aug-64	F	No	2	THURLSTONE	7
21	I	YOUSLY	11-Oct-75	M	Yes	2	SLKSTONE	1
22	.	POWERS	30-Aug-45	M	Yes	0	HENDON	123
23	.	MILSTER	15-Mar-60	M	No	0	DARBUSLY	J
24	.	MARSHALL	03-Jul-60	F	No	2	THURLSTONE	7
25	P	ARNOTT	10-Jul-65	F	No	1	SILFHILL	14
26	W	PURDJM	01-Sep-69	M	Yes	0	SHEFFIELD	14
27	U	MALIK	24-Sep-60	M	No	0	SILFHILL	14
28	A	OLIVER	24-Feb-68	F	Yes	1	SHEFFIELD	14
29	I	JAMISON	21-Jan-29	F	Yes	0	SLKSTONE	1

Record 1 of 41

Figure 4.3: Datasheet Form Layout



Notes

SREF	INIT	SNAME	DOB	GENDER	FES	KID	HTD/wt	Distance
J		OSMAN	29-Sep-23	M	M	0	MILLHOUSE GREEN	

Record: 1 of 41

Figure 4.4: Justified Form Layout

### Adding a calculated field to a form.

Forms are often used to display information that is not stored directly in data tables. The current age of a student is a good example of such a “calculated field”. Adding a calculated field to a form is identical to adding a calculated field to a report.

We are going to add a calculated field to the “All Students - Personal Details” form that displays the age of the student. If you refer to the worked examples in the Reports section of this course you will note that the functions used to calculate current age were the DateDiff() function and the Now() function.

To recap:

Now() returns today’s date.

DateDiff(<unit>,<date1>,<Date 2> ) calculates the difference between two dates.

The calculated field will contain the following:

**=DateDiff(“yyyy”, [DOB], Now())**

This will calculate the difference between the student’s date of birth ([DOB]) and today’s date (Now()). The unit of measurement will be years (“yyyy”).

Remember that any calculation will begin with an = sign.

The amended form is shown in Figure 4.5.



Notes

Figure 4.5: Form with a calculated field

To add the calculated field , work through the following steps.

- ☞ Ensure that the “All Students - Personal Details” form is open and that you are in design mode
- ☞ Widen the window to accommodate the new field
- ☞ Widen the form background by moving the cursor to the right edge until the cursor changes to a vertical line and a horizontal double headed arrow (see Figure 4.6). Drag the mouse rightwards to widen the form.



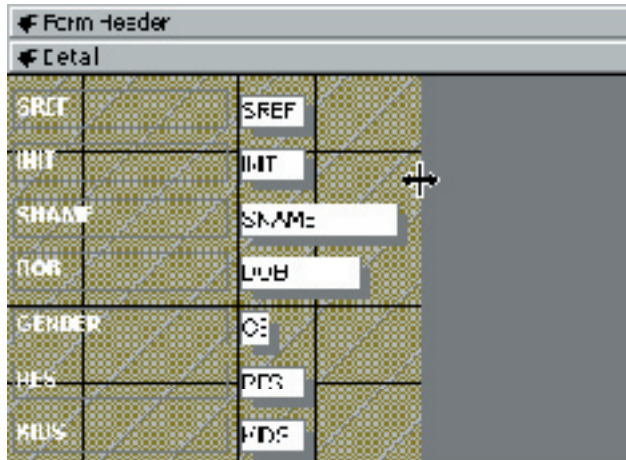


Figure 4.6: Widening the form

- ☞ Select a text box from the Toolbox and add it to the form just to the right of the DOB text box (which is bound to the DOB data field in the Table STUDREC).
- ☞ Adjust the positions of the accompanying label and the (unbound) text box. This can be achieved by moving the mouse pointer to the large handle in the top left of each object (the pointer will turn into a hand - see Figure 4.7) and dragging the objects to the desired position.
- ☞ Change the text in the label to “Age:”

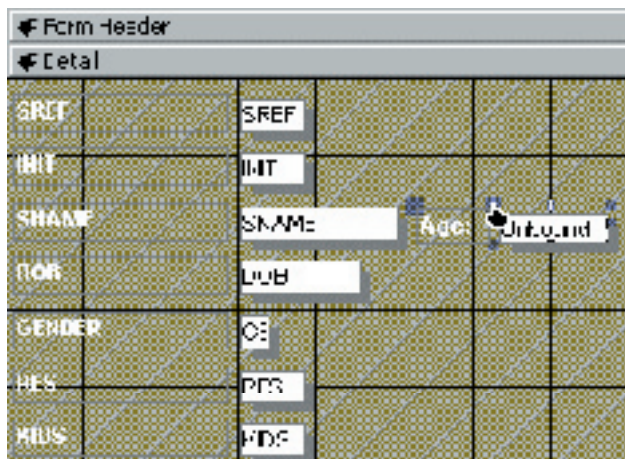


Figure 4.7: Adding and repositioning a text box.

- ☞ Right click on the unbound text box to access a pop up menu that has the option to change properties in it



- ☞ Select the “Properties” option
- ☞ Select the DATA tab
- ☞ Enter `=DateDiff("yyyy",[DOB],Now())` into the CONTROL SOURCE (the control source is where the data comes from), see Figure 4.8.

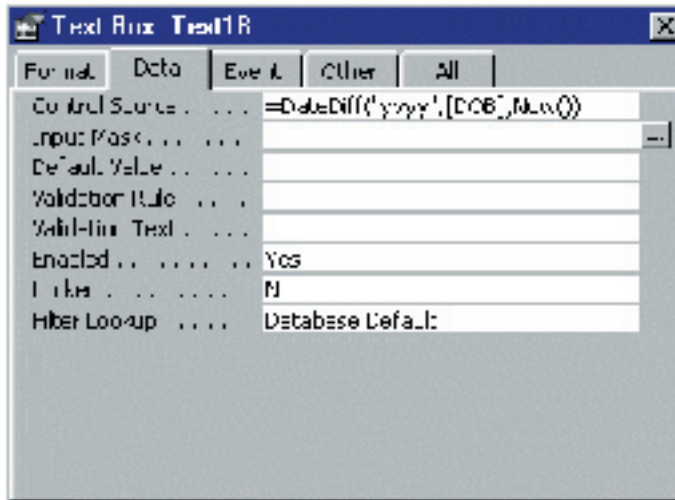


Figure 4.8: Defining the Control Source

- ☞ View your form in Form View, if it's OK save the changes.

### ***Using a Form to Add, modify and delete a record.***

Forms can be used to modify the data held in a table (or tables). It simply requires the user to overtype the contents of the text boxes in the form (remember that - apart from the age of the student - all the text boxes are bound to the underlying record structure of the table STUDREC).

Adding a record is done in exactly the same way as it is in the Data Sheet view. Move to the End Of the File (EOF) and fill in the blanks.

Deleting a record is done by selecting the record to be deleted and clicking on the “Delete Record” Icon.



## Restricting Editing Options in a Form

It is often the case that we require users to be able to view the data but not be able to change it in any way. This can be achieved relatively straightforwardly by changing the relevant properties of a form

We can determine whether users can Add records, Edit existing records and Delete records in any combination. This is done by viewing the form in design view and right clicking in the grey square in the top left of the form window (Figure 4.9: Form Properties Menu).

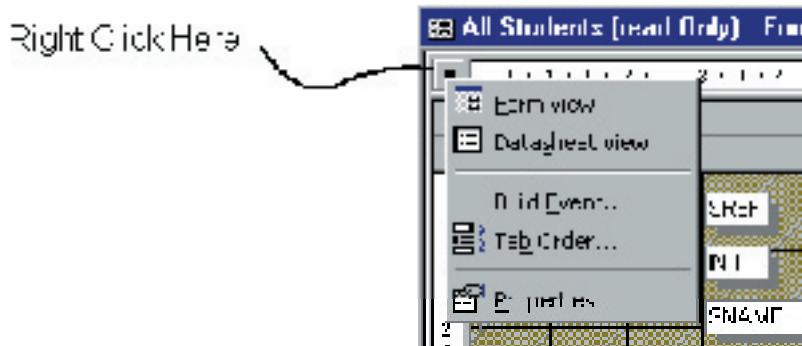


Figure 4.9: Form Properties Menu

Selecting the Properties menu will display the form properties window (Figure 4.10). The DATA tab will display the properties. It can be seen that the relevant properties are:

Allow Edits	Can Records be edited?
Allow Deletions	Can Records be deleted?
Allow Additions	Can Records be added?
Data Entry	If set to yes, only new records can be added with this form.

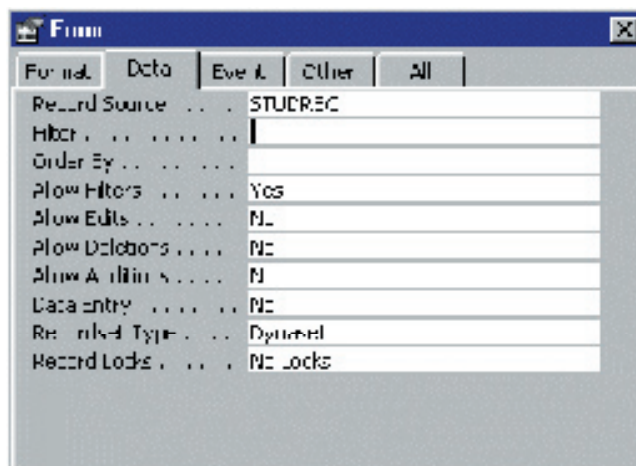


Figure 4.10: The DATA properties of a form



## *Exercise.*

Using Cut and Paste techniques copy the "All Student - Personal Details" form and paste it back into your database as "All Students (Read only)".

Amend the relevant properties of the new (read only) form so that no data changes whatsoever are possible.






Check that your form works by trying to add, delete and amend data - you shouldn't be able to do so.

**Do This NOW before continuing.**

## *Locking individual fields in a form*

It may be the case that we want to display all student details but allow only selected fields to be changed. For example, suppose that we wish to display a form that allows the user to change the RES field. We could do this by first of all setting the overall form properties to only allow editing of existing records (no additions, no deletions allowed).

In addition to that we could set each field (other than RES) to read only.

-  Cut and Paste another copy of the "All Students (Read only)" form as "Change Residential Status".
-  Open the "Change Residential Status" and ensure that you are in design view.
-  Change the properties of the form so that the ALLOW EDIT property is set to YES (note that the ALLOW DELETIONS and ALLOW ADDITIONS are both still set to NO).
-  Right Click on the SREF field to access a pop up menu
-  Select the PROPERTIES option of that menu (Figure 4.11).



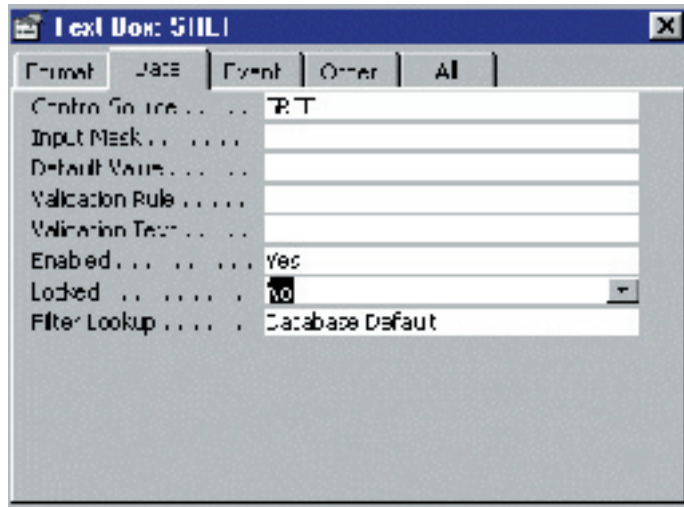


Figure 4.11: SREF Properties

- ☞ Change the **LOCKED** property from **NO** to **YES**. This will lock the field so that it cannot be changed in this form.
- ☞ Continue to do the same for the fields **INIT**, **SNAME**, **DOB**, **KIDS**, **HTOWN** and **DISTANCE**. Leave the field **RES** unlocked. Note: we don't need to do anything about **AGE** as it is a calculated field.
- ☞ Save your form
- ☞ View the form in "Form" mode and try to change the fields in record 1. You should find that the only field you can change is the **RES** field.
- ☞ Try to delete a record - you shouldn't be able to
- ☞ Try to add a record - once again, you shouldn't be able to.

**Refer to Exercise 4: at the end of this section and complete questions 1 to 5 before proceeding.**










Notes



## *Basing Forms On Queries*

As well as being based on a Table, a form can also be based on a query. This is useful if we wish to give a user the ability to change details of only a certain sub set of students. For example, we may wish to allow a recruitment tutor for Barnsley to edit details of only those students who live in Barnsley.

Work through this example.

-  Create a new view (using the form wizard) that is based on the query "BARNSELEY" and add all data fields to that view
-  Select a columnar layout and a "BLEND" background style
-  Save the form as "Barnsley - Edit Residential Details"
-  Edit the form details so as to prohibit additions or deletions of data records
-  Lock all individual fields except RES and KIDS
-  Save the changes
-  Run the query and verify that you cannot add or delete a record.
-  Also verify that you cannot change any data field except RES and KIDS (but be sure to reverse the changes you make)
-  Finally, note that only students who live in Barnsley can have their residential details edited in this way.

**Refer to Exercise 4: at the end of this section and complete questions 6 to 8 before proceeding.**



## *Creating a viable user interface*

The various parts of the student records database we have created to date are all capable of being used interactively, providing the user knows how to use Access. As database designers and developers, we may wish to present the work we have done to date in a more “user friendly” format. Until recently, this has meant that we need to resort to programming. However, with Access (and most other “modern” database management systems) it is possible to design a simple user interface by visual methods in exactly the way we have been working with forms in this section of the course.

The design of workable and efficient user interfaces is an important and increasingly well researched area that extends well beyond the scope of this course. This does not mean that you should dismiss these concerns; the following section is designed to get you thinking about exactly how easily a “user” could successfully interact with our student records system. Consider this list of “user rights” from usability expert Dr. Clare-Marie Karat, it’s a good starting point for your thoughts on User Interface design.

### ***User Rights***

**Perspective:** The user is always right. If there is a problem with the use of the system, the system is the problem, not the user.  
**Installation:** The user has the right to easily install and uninstall software and hardware systems without negative consequences.

**Compliance:** The user has the right to a system that performs exactly as promised.

**Instruction:** The user has the right to easy-to-use instructions (user guides, online or contextual help, error messages) for understanding and utilizing a system to achieve desired goals and recover efficiently and gracefully from problem situations.

**Control:** The user has the right to be in control of the system and to be able to get the system to respond to a request for attention.

**Feedback:** The user has the right to a system that provides clear, understandable, and accurate information regarding the task it is performing and the progress toward completion.



**Dependencies:** The user has the right to be clearly informed about all systems requirements for successfully using software or hardware.

**Scope:** The user has the right to know the limits of the system's capabilities.

**Assistance:** The user has the right to communicate with the technology provider and receive a thoughtful and helpful response when raising concerns.

**Usability:** The user should be the master of software and hardware technology, not vice-versa. Products should be natural and intuitive to use.

Karat, Clare-Marie *User Rights - the customer is always right*, IBM, [http://www-3.ibm.com/ibm/easy/eou\\_ext.nsf/Publish/12](http://www-3.ibm.com/ibm/easy/eou_ext.nsf/Publish/12) [Accessed 3rd October 2003]

## Using a form to create a simple customised user interface

Using Access, we can design a form that will have at least some of the characteristics of the preceding list. We can design a form that will act as a menu system that will shield the user from the technical complexities of the Access developer's interface. In other words, we can enhance the **Useability** and the **Control** that a non-technical user can be expected to experience when using our student records system.

Putting it simply, we can design a form that displays a menu of choices. All the user needs to do is to click on the required option.

This can be done by designing a form that is not bound to any data source; instead, the form has a number of "command buttons" on it which, when clicked on, will either open a form or run a report.

The finished form is shown in figure 4.12.



Notes

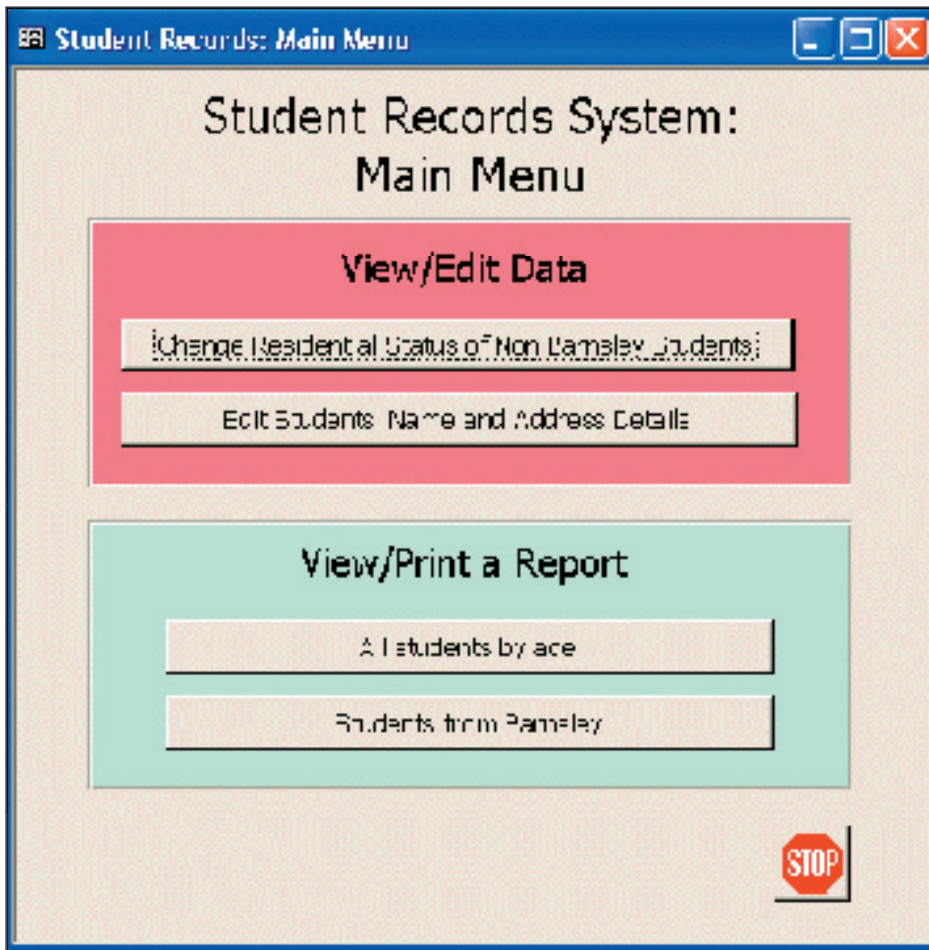


Figure 4.12: A simple user interface

The main menu shown in figure 61 represents a simple user interface that is intended to shield the user from the complexities of the underlying database management system. The main features of the form are:

- The main heading: informs the user whereabouts in the system they are.
- The “Stop” button: this closes the form, it is important to be consistent across all menus and sub-menus when using such graphical devices.
- The menu is clearly divided into two sections, “View/Edit Data” and “View/Print Report”
- The menu has four options (plus stop), two to run existing forms and two to run existing reports



## Constructing the Main Menu form

In this example we will use a range of new techniques, namely;

- Constructing a form using the design view (i.e. not using a wizard)
- Using graphics elements (a rectangle - two of them)
- Using Command buttons



Begin the process by ensuring that Access running and the Student Records database.



Select the Forms object in the Database Window and “Create a form in design view” (Figure 4.13).

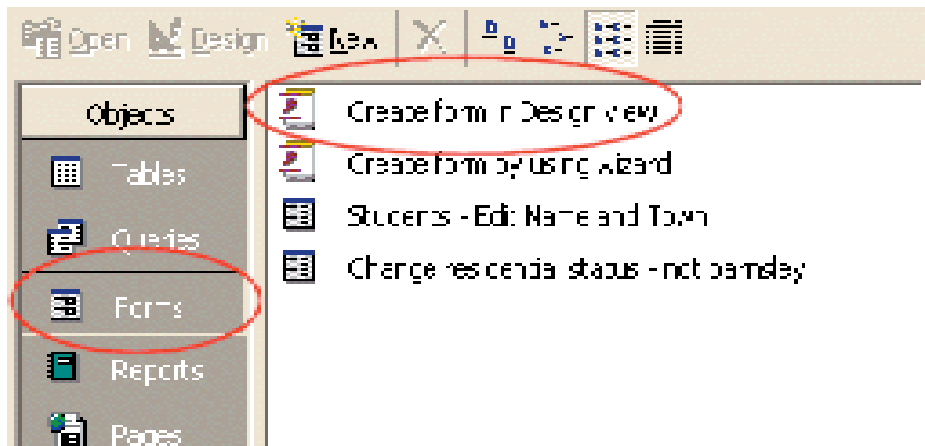


Figure 4.13: Creating a Form in design view

You should see on your screen a blank form (Form1) similar to that shown in figure 4.14.



Notes

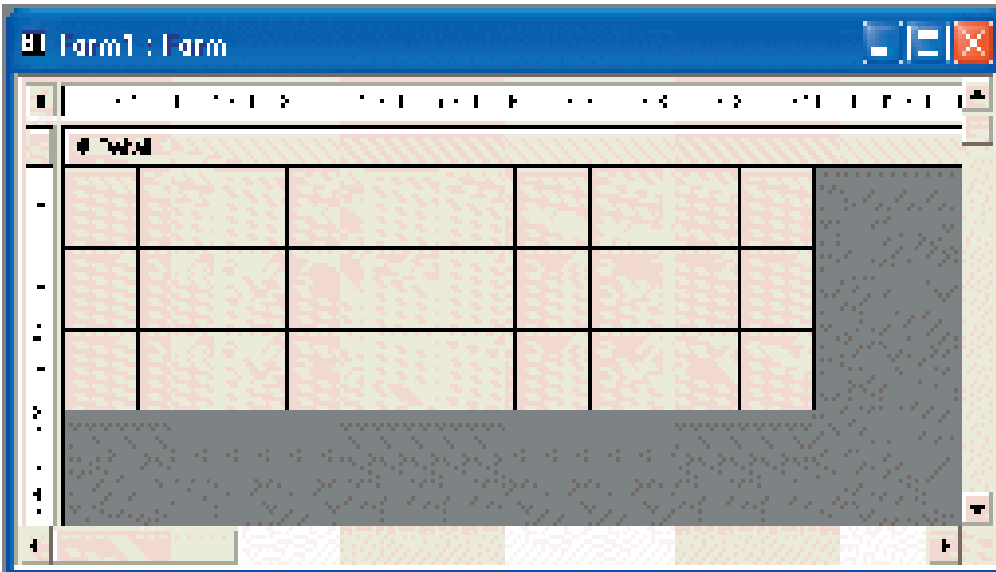
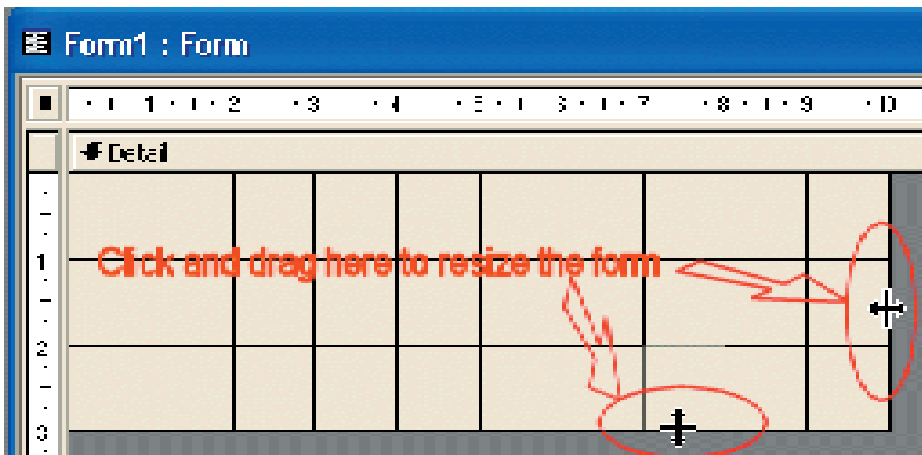


Figure 4.14: A new form in design view

Resizing the form is done by moving the mouse pointer to the edge of the form (the pointer will change shape) and dragging the required edge to the position you require; this is shown in figure 4.15.




Resize the form to the approximate shape and size you require (you can always change it later if you need to).





We now need to place our design elements on the form. This is done by making use of the “Toolbox” toolbar. We will use **Labels**, **Command Buttons** and **Rectangles** (Figure 4.16).



Figure 4.16: The Toolbox toolbar

- 

Select the Label tool and “drag out” the outline of the Main Menu Heading (again, don’t worry if it doesn’t fit exactly, you can always resize the label later).
- 

Type the text “Student Records System: Main Menu” (Hint: to move to the next line in a label, use the CTRL+Return key combination).
- 

Press the return key OR click anywhere on the form background with the left button of the mouse to complete the label text entry.



Notes

You screen should display a form similar to figure 4.17.

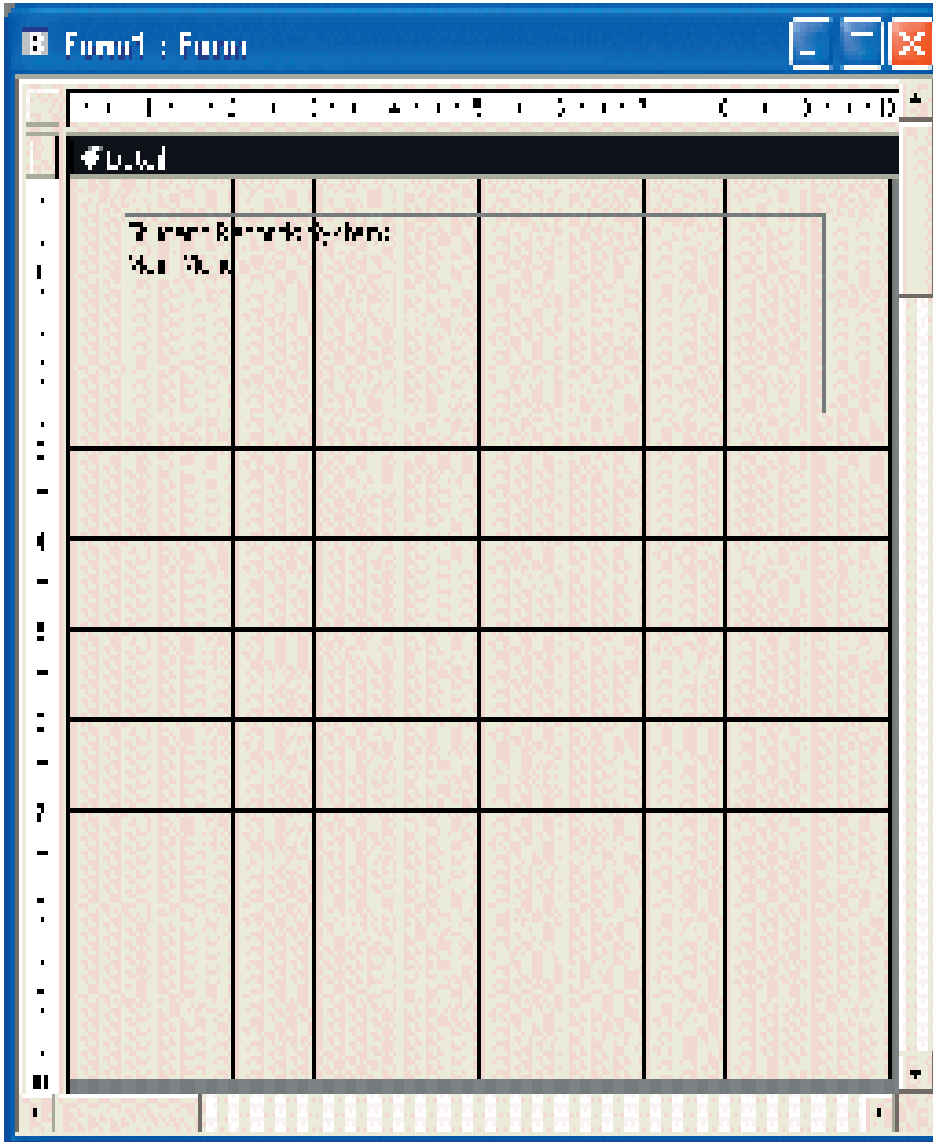


Figure 4.17: Main Menu Version 1

The label box needs to be formatted, the text needs to be larger, and it needs to be centered within the label.



Ensure that the Label box is selected and use the text formatting controls (figure 4.18) to format the text to Tahoma, 18 point and centered.



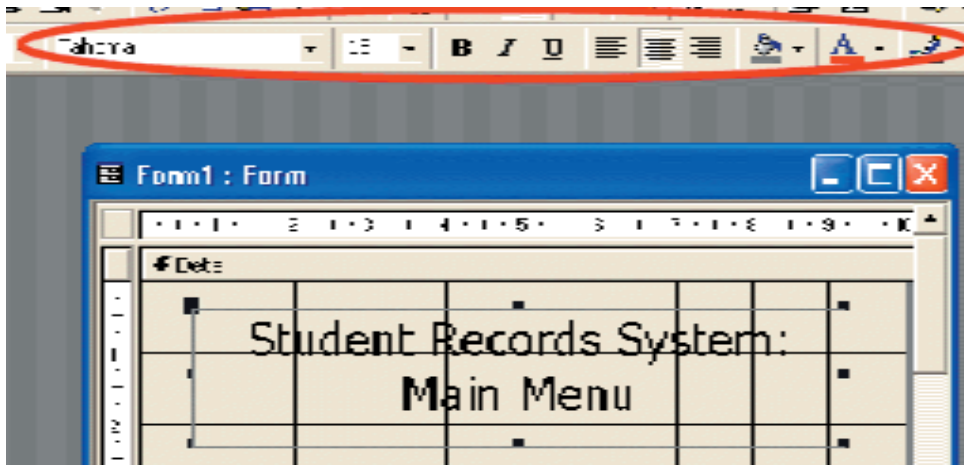


Figure 4.18: Main Menu version 2

☞ At this point Save the form as “Main Menu”

We now need to add two sections to the form to contain the “View/Edit Data” commands and the “View/Print a Report” commands. We’ll do this by drawing two rectangles on the form.

☞ Select the rectangle tool and draw two rectangles on the form.

☞ Change the “Back Color” properties of each rectangle so that the “View/Edit Data” section is a reddish colour (because it’s potentially dangerous to be able to change data) and green for the “View/Print Report” section (we won’t be able to change the underlying data from this section).

☞ Ensure that the “Back Style” property of each rectangle is set to “Normal” (as opposed to “Transparent”). This will enable us to see the colours of each rectangle.

Figure 4.19 gives an idea of the sequence of commands required.



Notes

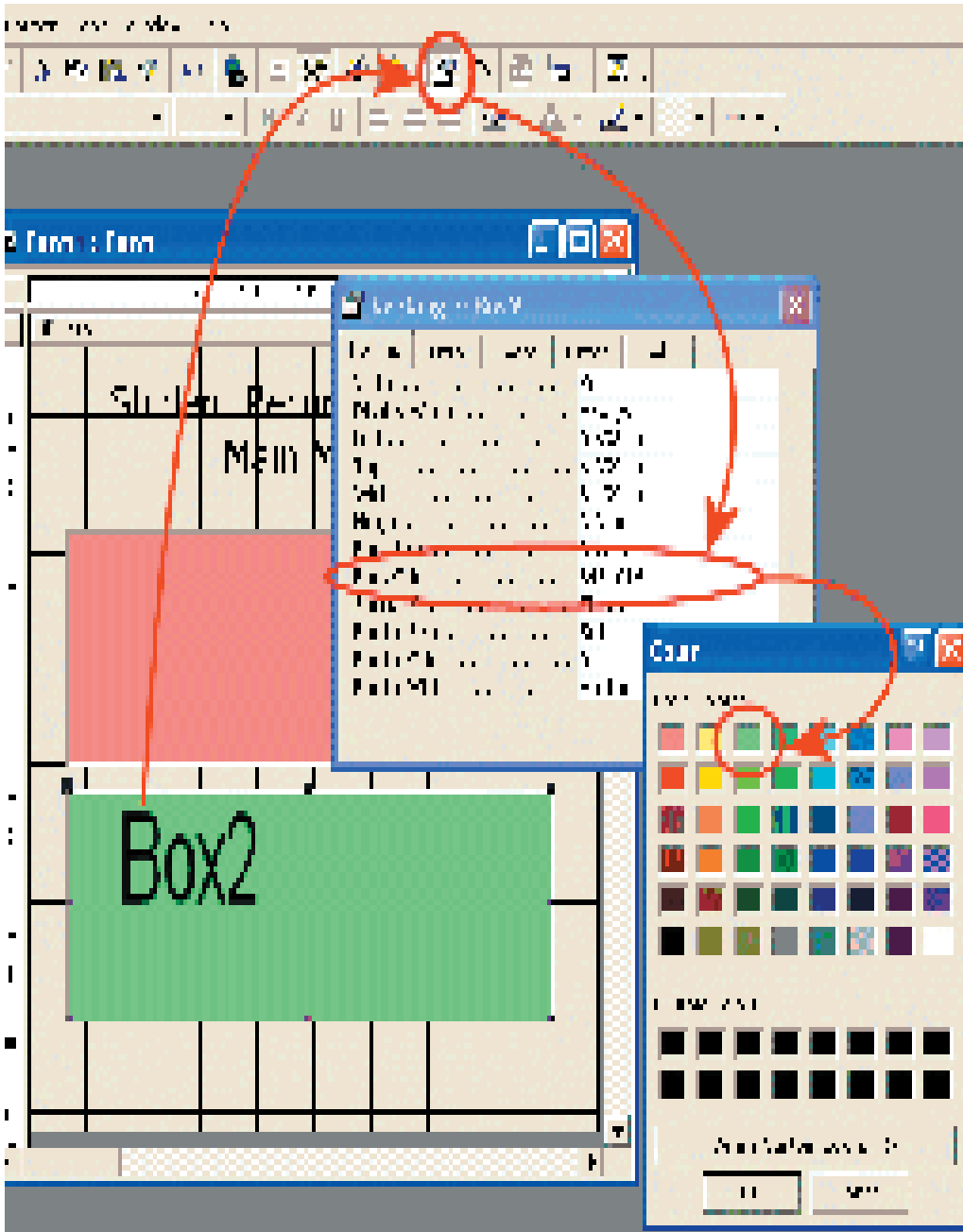


Figure 4.19: Changing the Background Colour of a rectangle

### Adding Command Buttons to the form

Command buttons are objects, which, when clicked on, will initiate an action. There are three actions we wish to initiate:

- Close the Main Menu
- Open another form
- Open a report



First, we'll place a command button in the bottom right of the form that will close the form when clicked on, this is in effect an "Exit" button.

- ➔ Ensuring that the form is in Design View, Select the "Command Button" from the "Toolbox" Toolbar.
- ➔ Move the mouse to the form and "drag out" the outline of the button in the bottom right of your form.



A "Command Button Wizard" will appear (figure 4.20).

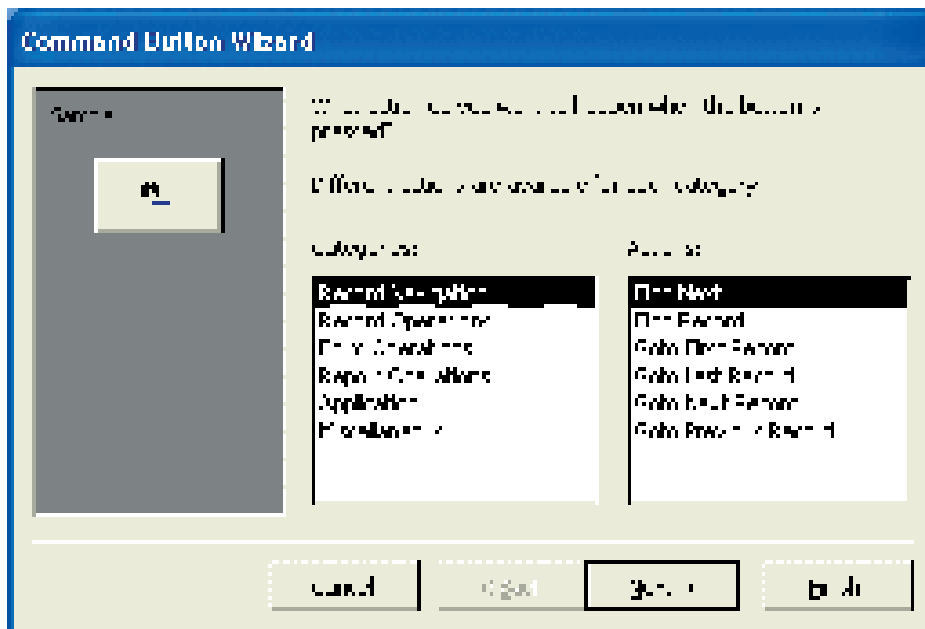


Figure 4.20: Command Button Wizard

- ➔ In the Categories window (left) select "Form Operations" and from the Actions window select "Close Form" (figure 4.21)

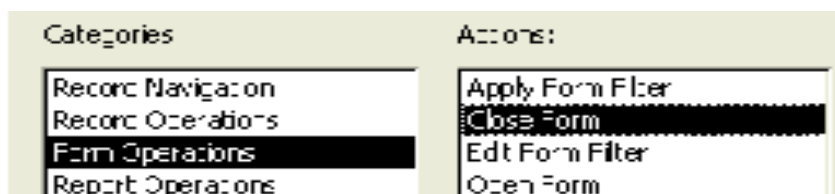


Figure 4.21: Close Form action

- ➔ Click on the "Next" button



- ☞ Select "Picture" and "Stop Sign" as shown in figure 4.22.

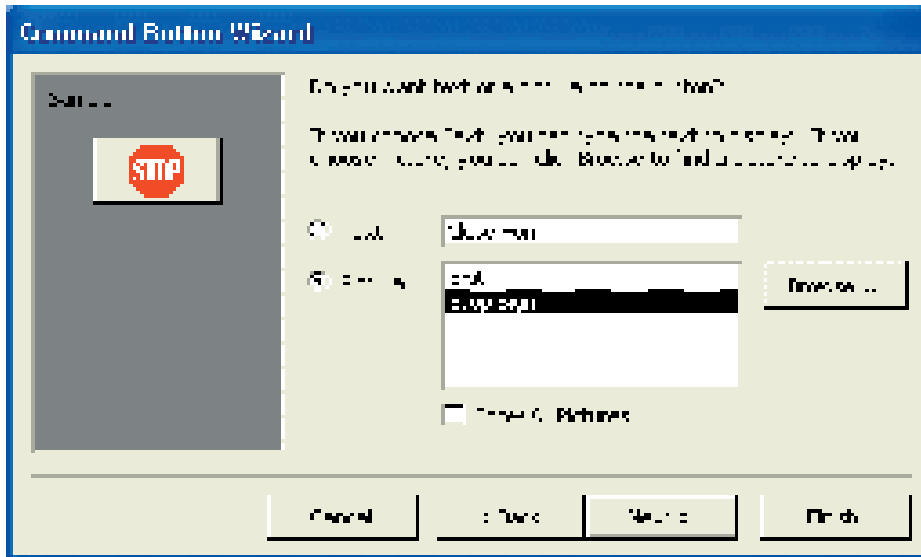


Figure 4.22: Assigning a picture to a command button

- ☞ Click on the "Next" button
- ☞ When prompted, give the command button a meaningful name (we'll call it stop)
- ☞ Click on the "Finish" button to complete the process.

You should now see a command button in the bottom right of your form. When the form is viewed in Form View, clicking on the stop button will close the form.

### Adding Command buttons to open forms and reports

We will now add a button to open a form and a second button to open a report.

- ☞ Select the "Command Button" from the "Toolbox" toolbar.
- ☞ Dragout the outline of a command button in the top (red) rectangle as shown in figure 4.23.



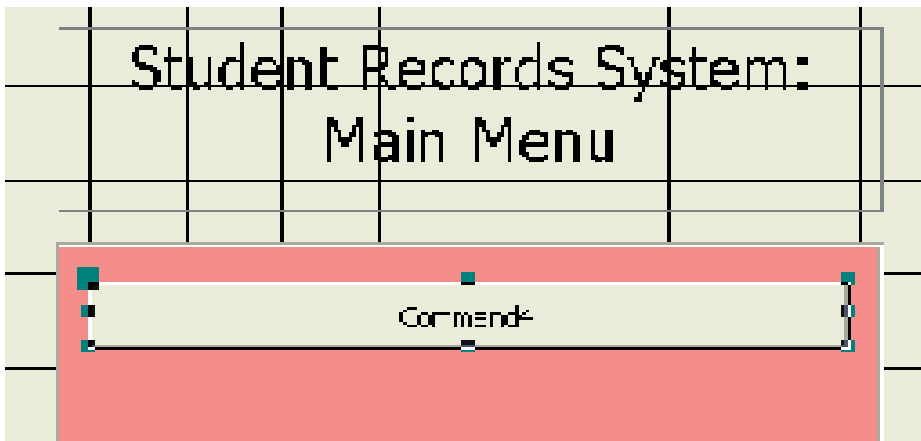


Figure 4.23: Command buton to open a form

- ☞ Select the Category “Form Operations” and the action “Open Form”
- ☞ Click on the “Next” button
- ☞ Select the form you wish to open (Change residential Status - Not Barnsley)
- ☞ Click on the “Next” button
- ☞ Select the “Open the form and show all the records” option (figure 4.24).

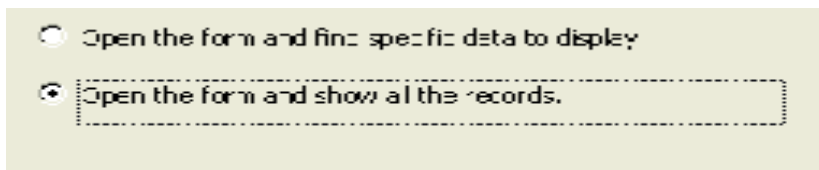


Figure 4.24: Show all records option

- ☞ Click on the “Next” button”
- ☞ Select the “Text” option and type “Change Residential Status of Non Barnsley Students” as the command button label (figure 4.25)



Notes

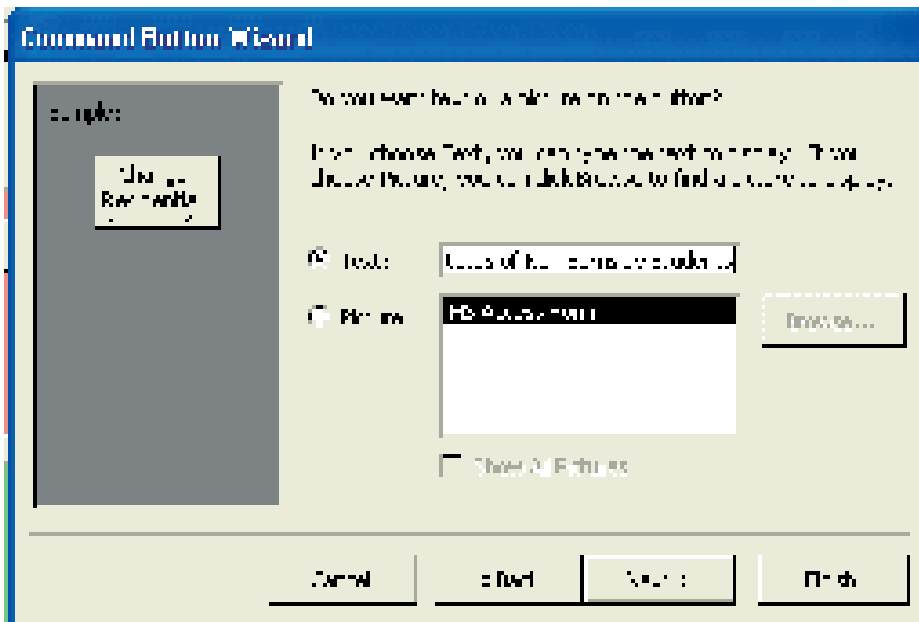





Figure 4.25: a command button with a text label

-  Click on the "Next" button
-  Rename the command button residential
-  Click on the "Finish" button

You should now have a button on your form, which, when clicked on will open up the required form (change residential status).

Placing a button on the "main menu" form to open a report is a similar process except that you need to select the category "Report Operations" and the Action "Preview Report" (**Don't select the action "Print Report" because this will send the report straight to the printer and we wish to conserve paper**).

Figure 4.26 shows the report choices you should make.



Notes

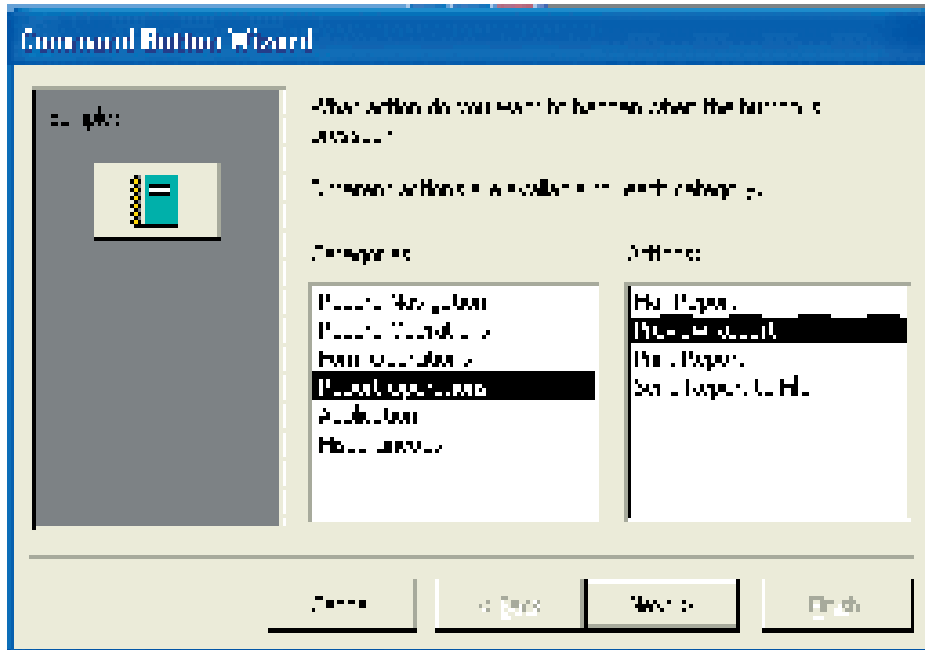


Figure 4.26: Previewing a report

**Exercise**

Refer to figure 4.12 on page 98 and complete the main menu form using the techniques outlined in this section.

Be sure to test your main menu form by running it in form view and testing the actions of each command button.

**Exercise 4:**

Answer all 9 questions.

For questions 1 to 8, use the "Print Screen" button and WORD to capture screen shots of the design view of each of your forms, save each answer to disk. For each question submit a screen shot of the relevant parts of the design view of the form, do this by pasting the screen shot into a word document. Ensure that each question is numbered and titled (e.g. Ex.4.1 "Add and Edit Records").

Send an e-mail to your tutor when you have finished the menu system required for question 9.

You will be required to demonstrate that your forms work to your tutor by showing the on screen versions.



**Question 1**

Design a form to allow a user to add and edit records. Display all of the Data fields. Use a Datasheet form layout.

Save the form as "Students - Full Edit"

**Question 2**

Design a form to allow a user to add and edit records. Display only the fields SREF, INIT, SNAME and HTOWN. Use a columnar layout.

Save the form as "Students - Edit Name & Town"

**Question 3**

Design a Form to allow a user to view (i.e. NOT change, add or delete) the following data fields SREF, INIT, SNAME, KIDS, HTOWN. Select a columnar format.

Save the form as "Students - Town & Children (Read Only)"

**Question 4**

Design a form to allow the user to view (on a read only basis) All of the fields in STUDREC. Your Form should contain a calculated field that displays the age of the student in years. Select a columnar format.

Save the form as "Students - Age (Read Only)"

**Question 5**

Design a read only form that displays all of the data fields in STUDREC. Add a calculated field that displays the word "LOCAL" is the student lives not more than 12 miles from the College, otherwise display the phrase "Not Local". Select Columnar layout.

Save the form as "Students - Local-Non Local (Read Only)"



**Question 6**

Design a form that is based on the query RESIDENT and allow the user read only access to all resident students.

Save your form as "Resident Students (Read Only)"

**Question 7**

Design a form to allow a user to edit only the RES field for those students who do not live in Barnsley.

Save your form as "Change residential status (not Barnsley)"

**Question 8**

Create a form that allows users to add records but not delete them or to edit existing records. Test your form by using it to add the following:

SREF=42 (assuming that you have already got records 1 to 41 in your table).

INIT= J  
SNAME=CHESTER  
DOB= 15/05/60  
GENDER=M  
RES=No  
KIDS=0  
HTOWN=BARNLSLEY  
DISTANCE=3

Save your form as "Add Student Details"

**Question 9**

Complete the exercise on page 108 (i.e. complete the "Main Menu" form).

