MICROSOFT ACCESS

A database is a collection of different types of data, stored in a manner to facilitate use in diverse ways. In **Microsoft® Access 2000**, a database contains Tables, Queries, Forms, Reports, Pages, Macros, and Modules. This handout provides an overview of some of the key features of Access that make it possible to create and use a database developed using this software.

A. CREATING A DATABASE

Start Microsoft Access, select the New option in the File menu, and Database from the General tab. The system will prompt you to provide a name for the database. You can select the directory where you wish to save this database. The default extension of an Access database is **mdb** (you can use any other extension if you wish). Once you have created a database, it will show up as a file in the directory you have specified. Please note that it is not a data file in the conventional sense (i.e., consisting of records of a particular type of data). Instead, it includes, in addition to such data, many other kinds of information that we will discuss.

B. CREATING TABLES IN A DATABASE

All data reside in tables. Before you can store data in a table, you must first provide the Table structure. Select Tables in the Objects list and click on it. Before you have created any tables, the list box will show three options. Choose "Create table in Design view" to create a table (other options are "Create table by using Wizard" and "Create table by entering data"). A window will appear with three columns: Field Name, Data Type, and Description. You will enter, in each row, details for a field of the table being created. Once the table structure has been specified in the Design View, you can then use the *Datasheet View* to see the contents of the table, and add or delete records from the table.

For example, consider the following table EMPLOYEE:

EmployeeId	FirstName	LastName	HireDate	Area	Married	No_of_ Children
234-15-4231	Chris	Date	1/4/88	MIS	Yes	2
321-45-6781	Steve	Jobs	8/17/90	MIS	Yes	3
876-54-4334	Scott	McNealy	8/14/87	OM	No	0
547-23-8645	Bill	Gates	8/20/93	GBUS	Yes	0

The table structure for the above table could be:

Field Name	Data Type	Description
EmployeeId	Text	
FirstName	Text	Includes Middle Initial as well
LastName	Text	
HireDate	Date	
Area	Text	
Married	Yes/No	
No_of_Children	Number	

For each field, enter the name, data type and description. Spaces can be included in field names; however, it is recommended that you use them only if absolutely necessary. The following different data types are available as options.

Text	Alphanumeric Characters (Max size 255)

Memo	Longer text descriptions
Number	Integers and Reals
Date/Time	
Currency	
Counter	System generated counter that is automatically incremented
Yes/No	Boolean data
OLE Object	Graphics, etc.

By default, the system assigns each field to be of type Text of length 50 characters. You can override the default option by clicking on the down-arrow button in the Data Type field in the appropriate row, and selecting the correct type. Once the type is selected for a field, you can specify additional field properties in the bottom half of the screen. Note that the displayed Field Properties are context sensitive.

SETTING FIELD PROPERTIES

Field Size:
This property is available for the data types Text and Number. When the data type chosen is Text, you are allowed to specify any length between 0 and 255. For instance, you may specify the field *LastName* to be of type Text and Length 15 characters.
If *Number* is chosen, then the options include Byte, Integer, Long Integer, Single and Double.

Byte	Allows integers in the range [0..255]
Integer	Allows integers in the range [-32768..32767]
Long Integer	Allows integers in the range [$-2^{31}...2^{31}$]
Single	Allows real values with up to 7 places of decimals
Double	Allows real values with up to 15 places of decimals

In the above example, the field *EmployeeId* may be specified as Long Integer.

Format:
This determines how data gets displayed in tables. The types of format available depend on the data type. Some of the useful format options are:

Text:

>	Displays all characters uppercase
<	Displays all characters lowercase

Number:

General	4567.8
Standard	4,567.8
Currency	$4,567.80
Fixed	4567.00
Scientific	4.567E+03
Percent	.1 will show as 10%

Date:

General Date	1/29/96 10:55:00 AM
Long Date	Monday, January 29, 1996
Medium Date	29-January-96
Short Date	1/29/96
Long Time	10:55:00 AM
Medium Time	10:55 AM
Short Time	10:55

Input Mask:
This determines how data is to be entered in tables. This is most useful with text fields.

0	Digit required
9	Digit optional
L	Letter required

?	Letter optional
A	Letter or Digit required
a	Letter or Digit optional
&	Any character or space required
C	Any character or space optional
#	Digit, +, -, or space optional
>	Character to the right converted to upper case
<	Character to the right converted to lower case

For example, the mask *(999)000-0000* will allow phone numbers to be stored with either 7 or 10 digits (note that if the area code is not known, spaces must be entered for those characters). The mask *>?<?????????* used with a 10 character text field will convert the first character to upper case and subsequent characters to lower case.

Default Values:
Allows specification of a default value for a field.
To specify the current date as the default date, enter **=Date()**.
To specify the current time as the default time, enter **=Now()**.

Validation Rules:
These rules allow us to validate the data before it is accepted by the database. Examples are:

> 0	All values in a field (e.g., *Prices*) must be greater than 0
<=Date()	Date entered must be less than or equal to today's date
>#1/1/96#	Date entered must be after 1/1/96
10 or 20 or 30	Acceptable values are 10, 20 or 30 only

Validation Text:
If the data entered by a user is invalid (wrong data type, or does not satisfy the validation rule provided), the system displays a generic message informing the user of the error. To help the user understand why the data entered is incorrect, we can specify, for any field, a customized message that will be displayed when a data entry error occurs for that field. For instance, we can include the message: "Hire Date entered cannot be after Today's Date!".

Caption:
Can specify a name for a field (different from the field name) which is displayed with the data in Datasheet View.

Primary Key:
A primary key is one or more fields in a table, the values in which uniquely identify each record in the table. For example, the SSN might serve as a primary key for a table that contains data on employees of a company. Note that Primary Keys are not mandatory for a table; however, in most applications they are *strongly recommended*.
A field can be identified as a Primary Key for a table by highlighting the row in the Design View (click on the gray box to the left of the field in the design view), and then clicking on the Primary Key icon in the Toolbar. Access automatically creates an index for the primary key field(s). Of course, remember that when adding data to the table, no two rows can have the same values in the primary key field(s).
Note that a primary key for a table may comprise of two or more fields. To specify such a primary key, highlight all the fields (using the Shift key) and then click on the Primary Key icon. Note that Access requires that such fields must be adjacent in the design view when the primary key is being defined. However, after the definition, the fields may be rearranged by clicking on a field and dragging it to some other position (without affecting its primary key status).
To *undo the primary key status* for a field, highlight the field in design view and click on the Primary Key icon.

Index:
We can specify an index on any field of a table. To create an index on a table, open the corresponding table in design view. Choose *Indexes* from the *View* Menu. Provide a name for the index, select the field on which the index is to be created, and specify appropriate properties (e.g. sort order, nulls allowed, unique, etc.). Note that creating an index may add substantial overheads to your database during processing.

Saving the Table Definition
After all fields have been specified, use the Save option from the File menu.

Printing the Table Structure
Select the *Analyze* option from the Tools menu. Choose *Documentor* within *Analyze*. Select object type *Tables*, and check the table name(s) you need. Use the *Options* button to control the level of detail by selecting on the appropriate buttons (called Radiobuttons). Usually, the *Properties* for the table, and*, Names, Datatypes,* and *Sizes* for each field in the table are adequate.

ADDING DATA TO TABLES

There are multiple ways in which you can add data to your tables. One way to do this is to highlight the table name in the Table List, and click on OPEN. This brings up the *Datasheet View* of the table. You can start entering the data in the fields. The cursor indicates where the data is being entered. The Tab key allows you to advance to the next field. The Datasheet View allows full screen editing. Note that if you enter a value that is illegal for a particular field, the system will show a message (default or customized). Each time you advance to the next record, the previous record gets *saved automatically*.

To delete records from a table, you can point the mouse to the gray box on the left of the record and click on it. This will highlight the entire record. Then select the Delete option from the Edit menu.

Printing the Table Contents
You can preview the table contents by highlighting the table in the Table List and using the Preview icon in the Toolbar. Use the Print option in the File menu to print. You can also execute the Print option using the pop-up menu that you get by highlighting a table in the Table List and clicking on the *right mouse button*.

MODIFYING THE TABLE STRUCTURE

Access allows a table structure to be modified at any time. For instance, users can add a field to an existing table that already has data in it. Users can also delete fields from existing tables; of course all the data in that field is lost in the process. Properties of fields can also be modified. Care must be taken that the new properties do not lead to invalidating data that is already in the table.

C. QUERYING A TABLE

A query is a question asked by a user to display selected data from a database. Queries can be based on single or multiple tables. Queries help users view only the relevant data from large databases. For instance, users may choose to view only some of the fields in a table, or, view selected records from a table, or selected fields from selected records, etc.

SINGLE TABLE QUERIES

Select Queries in the Objects list and click on it. Choose "Create query in Design view." MS Access will provide you with a list of tables to include in the query; the query will be based on the tables that you select. (Note that Access allows you to build a query from another query as well. The procedure is similar to that for queries based on tables.) Highlight the table(s) from which you wish to display data, and click on the *Add* button. Close the list of tables. The selected table will appear with all its fields in the Select Query window. The bottom half of this dialog box is called the **QBE** (Query By Example) grid.
To indicate the fields to be included in the query, highlight a field from the field list and drag it to a column in the QBE grid. Alternatively, you can double click on a field to select it. You could also use the pull-down arrow in the Field row in any column of the QBE grid to select a field!

Viewing a subset of fields
Select the desired fields in the QBE grid. One way to view the data in the selected fields is by clicking on the *Datasheet View* icon in the Toolbar. Alternatively, you can run the query by clicking on the *Run Query* icon (!) in the Toolbar. The selected data will be displayed on the screen.

For example, you may wish to view only the *HireDate* for each employee from the Employee Table. This is accomplished by selecting the corresponding fields in the QBE grid and running the query. The displayed data will look as below:

First Name	Last Name	Hire Date
Chris	Date	1/4/88
Steve	Jobs	8/17/90
Scott	McNealy	8/14/87
Bill	Gates	8/20/93

To return to the *Design View*, click on its icon in the Toolbar. You can choose the order in which fields are displayed by highlighting a field in the design view, and then dragging it to the desired location.

In MS Access, the data selected as a result of a query is called a dynaset. In some cases (not all), data can be added to the original underlying table (sometimes referred to as the *base table*), or modified in the table, from the dynaset.

Printing the results of a query
You may print the result of the query by clicking on the *Printer* icon in the Toolbar. Sometimes, you may wish to save a query for later use (if it is to be used often). To save the query use the *Save* option from the File menu and provide the query an appropriate name. It is a good practice to provide meaningful names to queries as much as possible.

It is important to note that the result of a query is dynamic. This means that the result depends on the actual data values stored in the underlying tables. When that data changes, the result of the query reflects those changes. Therefore, saving a query really means that we are saving the definition of the query, and not the displayed data that results from running the query.

Viewing a subset of records

Users can specify, in the QBE grid, the criteria for selecting records to be displayed. For instance, users may wish to see details of all employees in the area of MIS. The following selection criteria would be entered in the Area field:

=“MIS”

Note that the field used to select records *need not be displayed in the result* (e.g. if all displayed data correspond to employees in the area MIS, then there may not be any need to display that field at all!). Only fields with option *Show* checked are eventually displayed.

The field name to be displayed with the results of a query can be renamed within a query if desired. For example, the field name *FirstName* may be replaced by the label *Employee First Name* in the result of the query (without affecting the underlying table itself). In order to do that, enter the following in the corresponding field box:

Employee First Name: [FirstName]

Some other commonly used selection criteria are:

i. Field values greater than a specified one

e.g. in the No_of_Children field: > 1

ii. Field values in a specified range

e.g. in the No_of_Children field: >=1 And <=3

iii. Conditions on two separate fields

e.g. employees hired prior to 1/1/95 in MIS
in the Hire Date field: < #1/1/95#
And
in the Area field: =“MIS”

To select those records that satisfy both criteria, the conditions must appear in the same criteria row of the QBE grid. To select those records that satisfy any one of these two conditions, each condition should appear in a separate criteria row.

iv. Multiple conditions on the same field

e.g. *Area* is either “MIS” or “OM”
in the *Area* field: = “MIS” or “OM”

Access allows a query to have multiple conditions in the same field *simultaneously* with conditions in multiple fields.

v. Using wildcards for pattern match

e.g. in the Last Name field: G* selects all last names starting with G
e.g. in the Last Name field: G??? selects last names starting with G and length 4
e.g. in the Hire Date field: */*/94 selects all dates with year=94

vi. Sorting data based on a specified field

Use the pull-down arrow in the Sort field to specify the desired sort option (i.e. ascending or descending).

vii. Creating new calculated fields

Expressions can be entered in unused field cells in the QBE grid. For instance, if you have two fields, Price and Quantity, you can enter the following expression:

Amount:[Price]*[Quantity]

When the query is run, the new column has the label Amount, and the values displayed in each row are the products of the corresponding values in the Price and Quantity fields respectively.

Interestingly, Access allows calculated fields that are the sum of two text fields. The result is a concatenation of the contents of the individual text fields.

viii. Formatting numeric data during display

Numeric data can be displayed in appropriate formats in a query. To do this, highlight the field, and click on the right mouse. Select the *Properties* option, and then click on the list box next to the *Format* option to complete your selection. For example, you can choose the *Currency* option to display a field *Prices*.

ix. Parameter queries

A parameter query allows the user to specify the value used in a selection criteria when the query is executed. For instance, you may wish to display the names of all employees who work in the *Area* MIS, whereas at another times you may wish to display the names of employees in OM or some other *Area*. One alternative is to design a separate query for each of the different possible *Area*s. Alternatively, a parameter query could be used that will prompt the user to specify the appropriate *Area* when the query is run, and then select records according to the input provided by the user.

To create such a parameter query, select all the fields to be displayed (e.g. FirstName, LastName, etc.) and the condition field (which is *Area* in this example). In the condition row of that field, enter within square brackets the message you wish to be displayed to prompt the user (e.g. [Enter the Area name:]). Deselect the Show option if the *Area* field is not to appear in the result. When the query is run, the user will be prompted to enter the *Area* name. Based on the user's response, the corresponding records will be selected and displayed.

A parameter query may have parameter conditions specified on several fields. Also, multiple conditions may be parameterized on the same field. For instance, employees hired during different periods of time may be displayed using a parameter query where the user provides the start and end dates. The parameter query would have the following condition for the *HireDate* field:

>=[*Enter Begin Date:*] AND <=[*Enter End Date*:]

x. Totals in queries

A query can be used to provide the totals of values in a field. Note that you cannot display both the details and the total in the same query (however, you can do that in a report in Access). If you wish to display the total number of children for all employees in MIS, you have to first select the field *Area* and *No_of_Children* in the QBE window. Highlight any one field and click on the right mouse button. Select the *Sum* option. Doing this will add a row called *Total* in the lower half of the QBE window. Use the drop down arrow in each field (on the Totals row) to specify the appropriate requirement. For field *No_of_Children* select *Sum*, and for *Area* select the *Where* option. Specify the desired *Area* in the Criteria row and run the query.

If you wish to obtain totals for each *Area*, you will need to specify, for the *Total* row in the *Area* field, the *Group By* option and then run the query.

MULTIPLE TABLE QUERIES

Consider a second table COURSES that has the following data.

COURSES

Course Name	Department	Instructor Id
Intro to MIS	ISDS	234-15-4231
Data Structures	ISDS	321-45-6781
Data Bases	ISDS	876-54-4334
Systems Analysis	ISDS	547-23-8645

Users may wish to know the name of instructors for each course. Since that information is available in the Employee Table, users can design queries that will combine data from the two tables before display. Such an operation is called a **Join**.

Select the Query tab in the Database window and Click on New to create a query. Select *Design View* from the resulting list box. Access will provide you with a list of tables to include in the query. Highlight the tables EMPLOYEE and COURSES, and click on the *Add* button (you can either add tables one at a time, or highlight both using the *Ctrl* option and add them in one operation). Close the list of tables. The selected tables will appear with all its fields in the Select Query window.

An important step in matching data from the tables is to specify which field of one table should match which field of the other. In this example, this may be done by highlighting the *EmployeeId* field in table EMPLOYEE, and keeping the mouse pressed, dragging the cursor to the *InstructorId* field in table COURSES. A line connecting the two tables will show up. Add desired columns from the two tables into the QBE grid as before. Save and run the query.

In some instances, Access can automatically figure out which columns should be joined. This happens when two fields in two separate tables have been given the same name and data type, and one of them has been defined as the primary key (in its table).

D. USING FORMS

Forms allow users to easily maintain data in tables, and view data from tables one record at a time. They are usually used for data entry screens and creating electronic documents that resemble traditional paper-based business documents (e.g., purchase orders, invoices, etc.). Before creating a form, it is useful to design the form on paper. Designing a form involves identifying different items that would appear in the form, the layout for the form, how the items are to be displayed (formats, fonts, color, etc.), and the source of data for these fields (user input, existing tables, results of queries, etc.). We first consider a simple form that is to be used to input data into a table.

You can create a Form either using the Design view, or by using the wizard. I will describe how a form can be designed from scratch using the Design view (the other option is fairly straightforward once you understand what goes into creating a form).

Select Forms in the Objects list and click on it. Choose “Create form in Design view.” A blank form template will be displayed. Usually, you will use a form to display fields from a table or a query. For such forms, you should first identify the query or table from which the form will get its data values. To do this, highlight the (blank) form object itself, click the right mouse button, and select Properties. A new window, named Form, will appear on screen. Use the drop-down list in the Record Source option of the Data menu in this new window to select the appropriate table (or query). The Field list will automatically appear on screen. Now, you can include fields in the form template by highlighting a field in the Field list, and dragging it onto the form template (the Field List option is inoperable until you first select the appropriate table or query). Both a *Label* and a *Text box* will appear in the form by default (more on Labels and Text boxes follow). Note that once you have specified the table or query for a form, you can always display the Field List by selecting it from the View menu.

You can increase the size of the form by dragging on its edges. A Toolbox should also appear on the screen. If it is not there, click on the Toolbox option in the View menu. Some of the icons in the Toolbox commonly used for designing forms are discussed below.

Labels

Labels are boxes in which you can type in text you wish to display on a form. For example, the label DATA ENTRY - EMPLOYEE TABLE may be displayed on top of the form. To do this, click on the Label icon in the Toolbox, move the cursor onto the form and click on the part of the form where it should appear. You can then type in the text to be displayed. To resize the label, click on the Select Object icon in the toolbox, and then click on the label to display its handles (you can drag on these handles to resize the label).

Each object has associated with it a list of properties. To display this list, select the object (e.g. a label) and click on the *right* mouse button. The properties list will show up. For instance, you can change the font type and size by clicking on the appropriate field in the properties list, and using the pull-down menu to select the desired options.

The color of labels can be controlled using the Color icon on the Toolbar. Foreground and Background colors can be specified using this option.

Data Fields (Text Boxes)

Data fields can be included in the form either by using the Field List option from the View menu, or the Text Box icon in the Toolbox. When you click on the Field List option, the list of fields for the table are displayed. Highlight a field and drag it onto the form. Two boxes will show up, one that serves as a label, and another that displays the data for that field. By default, the label shows the field name; you can change it by typing any text you wish. The field box is linked to the table, and will eventually display values from that field.

Any time you wish to see how the form looks like, click on the Form View icon in the Toolbar. Clicking on the Design View icon brings back the design mode.

To create fields on the form using the Text Box in the Toolbox, you must explicitly specify what the field will be linked to. Click on the Text Box, move the cursor to the form, and click on the desired part of the form. You can type in text for the label box as before. To link the Text box to a field in a table/query, bring up the Properties list (select the field box, click on the right mouse button), and then move to the Control Source field (under the Data tab). Use the drop-down list to select the field to displayed.

Colors for data fields are controlled as for Labels.

Calculated Fields
You can choose to display fields that will display some calculated value in the box. For example, a field can be used to display the amount each employee should pay to cover expenses for their children for a departmental picnic. Let an amount of $10.00 be paid for each dependent. Display the properties list for the calculated field. Enter the following expression in the Control Source field of the properties list:

=10*[No_of_Children]

The Format can be set to Currency in the properties list.

Check Box
A check box is useful for displaying data items that are of type boolean (Yes/No). To display a Check Box, click on the icon in the Toolbox and place it on the form in the desired place. Enter text in the associated label box as before. Use the properties list to link the Check Box to the desired field (using the Control Source field in the Properties list).

The Option Button also works in a similar manner.

List box
The list box is used to provide users a list of values from which they can select one. For example, the value for field Area may either be MIS, STAT, or OM. Click on the List Box icon and place it on the form. A list box wizard will then guide you in specifying the values that should be displayed to the user when the form is in use. One common requirement is to provide in the list box those values that exist in some field of a table in the database (e.g. only those areas that already appear in the Area field). The other often used alternative is to enter a set of values in the Row Source Type and Row Source properties of that list box (this can be either accomplished through the wizard, or directly by editing these properties). For example, you may do the following : (i) Specify the field Area for the Control Source; (ii) Select Value List for Row Source Type (iii) In the Row Source field, specify: “MIS”; “OM”; “STAT”. When this form is used for data entry, a drop down list will be presented to users to select an appropriate value from these choices.
You can type in the label as desired.

Combo Box
The Combo Box is very similar to the List Box. The main difference is that users can input values other than those provided as part of the drop down list.

Printing a Form
Open the form to print. If you use the default Print option, Access will print the contents of each record in the underlying table/query using the form structure. Access also fits as many records in a page as possible, which is sometimes undesirable. Usually, you will be interested in printing information from a single record. First, locate the desired record, using the Find option in the Edit menu. For example, if you wish to find the record for an employee with *LastName* = “Gates”, you could place the cursor in the *LastName* field, invoke the Find option, and then enter the value “Gates.” Access allows you to locate multiple records with the same value. Once the desired record is located, choose Select from the Edit menu, invoke Print from the File menu, and specify range Selection.

Read Only Forms
Sometimes, users should be allowed to view records using a form, but not allowed to edit the data. For this type of use, open the form in Design view. Choose Select Form from the Edit Menu, and then Properties from the View menu. Change the Data Entry and Allow Edits properties to No. Save the form.

The Format menu allows different boxes to be aligned.

Sub-forms within Forms
In some applications, you may need to display data items from more than one table within a single form. This is usually needed when two tables have one common attribute (at least), and you wish to display a record in one table

along with all corresponding records in the other table based on the common attribute(s). Such forms could be used either to enter data into the two tables, or for viewing existing data from those tables (or both).

You can use sub-forms embedded within other forms to perform such tasks. I will describe how you can create a form with a sub-form for entering data into two related tables. Typically, a record in one of the two tables will have several records that correspond to it in the other table, but not vice versa (i.e., there will be a 1-to-many relationship across records in the two tables). The table on the one side of the relationship is called the parent table, and the other table is called the child table. For example, assume that each course in the COURSE table has only one instructor, and it is possible that an employee can teach several courses. In that case, the EMPLOYEE table is the parent table, and the COURSE table is the child one (this is because an Employee ID in the EMPLOYEE table can correspond to several Instructor ID's in the COURSE table, but not the other way around).

Create a form for the main table (EMPLOYEE, in our example), including all the fields in the form (we will refer to this form as the main form). Then, click once on the Subform/Subreport icon in the Toolbox, move the cursor over to an appropriate position in the main form and click on it. A template for the sub-form will appear within the main form, and MS ACCESS will automatically launch a wizard that will prompt you to provide the necessary details for the sub-form. First, you will be asked to indicate whether the sub-form is to be created for an existing table/query, or whether you wish to use a sub-form that has already been created. Assuming you do not have an existing sub-form, you can select the other option. The wizard will display a screen that will allow you to select the appropriate child table (COURSE, in this example), and then indicate the fields from this table you wish to include in the sub-form. Next, you will need to specify the fields in the two tables that are used to link the data in the two tables (in our example, Employee ID in EMPLOYEE, and Instructor Id in COURSE). You will be prompted to save the sub-form. You will be able to see the main form with the embedded sub-form in design view, as well as, use the form view to see how it will look in practice. You can now save the main form. Note that the main form and the sub-form will show up as separate objects in your database. You will find that when you enter data into the parent table using this form, the linking field in the child table will automatically inherit the value of the corresponding field in the parent table.

Note that you can modify the sub-form to change the way it looks. First, you can customize the sub-form to make the size and appearance compatible with the main form. You may also change the way data is displayed in the sub-form. By right clicking the top-left corner of the sub-form in the Design View of your main form, selecting the Properties option, and moving your cursor to the Default View in the Format tab, you will be presented several options.

You can decide to create your sub-form beforehand, and then insert it into the main form. When creating the main form, you will use the Subform/Subreport icon in the Toolbox (as discussed above). Choose the appropriate wizard option, and remember to identify the appropriate linking fields for the two tables (in this case, the fields will link the main form and the sub-form). Save the main form.

E. REPORTS

Access allows users to prepare reports based on Tables and Queries. The simplest form of a report is from the *datasheet view* of a Table (or a Query). The view can be printed out using the Print option in the File menu. Note that no additional headers or footers, or totals of columns, can be provided in such reports.

Analogous to forms, reports can either be created from scratch (in Design view) or by using a wizard. Note that usually we need to include in a report data that exists in two or more tables. To do this, we must first create a query that includes all these data, and then use the query to generate the report. I will briefly discuss how to use the wizard, and then how to create a report from scratch.

USING THE REPORT WIZARD

The wizard helps the user design a report by providing some of the commonly needed features as options that the user can select. It suggests default report headers, and allows Group Totals to be provided in the report. To use the wizard, select Reports in the Objects list and then choose “Create report by using wizard.” The wizard first prompts the user to identify the table or query from which the report is to be generated, and then allows the user to select the fields to be included in the report (from the underlying table/query). Next, it prompts the user to specify if data needs to be grouped based on values of a particular field (e.g., by Area). Multiple levels of Grouping are allowed. Next, the wizard allows the user to specify sort orders. Finally, several report layouts and styles are presented to the user. Based on all these selections, the wizard generates the report. Note that the report prepared using the wizard can be modified later using the Design view.

CREATING REPORTS IN DESIGN VIEW

You can design a report from scratch using the “Create report in Design view” option when creating a new report. The design procedure is similar to that for creating a Form. When you initiate this process, Access displays the Design View for a blank report that is divided in three horizontal sections (called bands). The first band is the Page Header, the middle one is the Detail, and the last one is the Page Footer. You can customize each of these sections (bands) by adding labels, fields from tables/queries, calculated fields, etc. Whatever is specified in the Page Header band will be printed on top of each page of the report. Similarly, anything in the Page Footer will appear at the bottom. Typically, you will include items like labels (e.g., field names), page number (a field with the expression *=Page*) and the current date (a field with *=Date()*) in such bands. You can draw horizontal lines in the Page Header band using the Line icon from the Toolbox. Record details (i.e., field values) are specified in the Detail band.

In design view, once the blank report template is displayed, you should identify the query or table from which the report will get its data values. To do this, highlight the (blank) report object itself, click the right mouse button, and select Properties. A new window Report will appear on screen. Use the drop-down list in the Record Source option of the Data menu in this new window to select the query (or table). The Field list will automatically appear on screen. Now, you can include fields in the Detail band by highlighting a field in the Field list, and dragging it onto the detail band (the Field List option is inoperable until you first select the appropriate query or table). Both the field Label and the field Text box will appear by default. If the field name is to be specified in the Page Header band, then you may wish to delete it from the detail band (select the label object and hit the Delete key). You can position each field as desired, size and align fields, and specify fonts, font sizes, and colors for the background and foreground, in the same manner as for designing Forms.

Another way to add new fields to the detail band is by selecting the Text Box icon in the Toolbox and dragging it over to the Detail band. A label box and a field box will appear on the report. Select the field box (which should display *unbound* at this time) and display its properties (clicking on the right mouse button). You can then link this box to any field in the field list by clicking in the Control Source property and using the pull-down arrow to display the list.

Calculated fields can also be included in reports in a similar manner. For example, if there are two fields Price and Quantity, then you can specify in the Control Source property the expression *=[Price]*[Quantity]*. This will lead to displaying the corresponding product in that field.

Additional Bands

Additional bands can be added to the report as desired. Report Header (and Report Footer) bands are added to the report using the ReportHeader/Footer option in the View menu. When reports are long and cannot fit in one page, then both Report Headers and Page Headers are desirable. Report Headers will appear in the first page of the report only, and typically include the report name and other information that is unique for that report. The Page Header would typically include the field names.

Bands for Group Headers are specified by clicking on the Sorting and Grouping icon in the Toolbar (or by selecting the Sorting and Grouping option in the View menu). A field name must be specified for a Group in the Sorting and Grouping window that appears on the screen. This window allows the user to specify if separate bands are to be provided for each group of records. If they are to be included in the report, they will show up as a Group Header band and a Group Footer band (in design view) with names corresponding to the grouping field. For example, if *Area* is the field used for Grouping, the Group Header band will show the name *Area Header*. Often, totals of numeric fields must be displayed for each group of records. This can be displayed in the Group Footer using an expression such as *=sum([Quantity])* (assuming you have a field called Quantity).

Page Breaks

The Page Break icon can be used to indicate page breaks in the design view.

F. MACROS

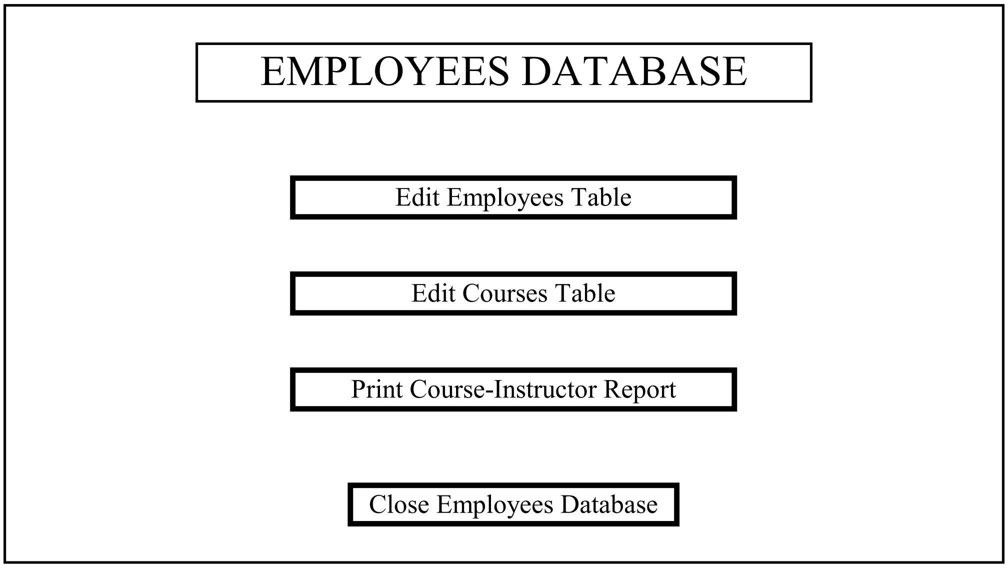
Macros are used to automate routine operations, and develop menu-driven systems. A macro is a list of actions that you can instruct the system to perform by clicking on a button or selecting a menu item. Macros are very useful for providing a customized interface to users of a database.

To create a macro, click on the Macro button in the database window, and select New. This will open a macro design window which is split into two parts: the upper part contains *actions* and *comments*, and the lower part displays *action arguments*. You can specify a desired action for a macro by using the pull down list in any field in the action column, and selecting that action. Some commonly used actions are:

- **OpenForm**: Opens and displays an existing form on the screen
- **OpenReport**: Opens and displays an existing report on the screen
- **OpenQuery**: Opens and displays an existing query on the screen
- **OpenTable**: Opens and displays an existing table on the screen

- **GoToRecord**: When opening a Table, specifies whether to display the First, Last or a New record in the table
- **Print**: Prints the active object (e.g. a report)
- **Close**: Closes the active window
- **Maximize**: Enlarges the selected window to fill up the entire screen

<u>Creating a Switchboard</u>
A commonly used interface for a database application is a form containing buttons. Consider the form shown below. The user can click on the different buttons to execute different actions. For example, by clicking on the Edit Employees Table button, the system can open the input form for the Employees table. After the table is updated, the user can click on a similar button in the input form, and return to this form. Similarly, the user can print out an existing report by clicking on the corresponding button in this form. Depending on the users requirements, the report may be previewed before the user actually prints it, or it may be automatically sent to the printer. Such actions are specified in the corresponding macros.



To enable the system to perform the desired actions, these buttons must be linked to macros. An easy way to design a switchboard menu of the kind shown above is as follows. First, design the necessary tables, forms, and reports (and associated queries if necessary). Next, create a macro for each action (or set of actions) that you wish to link to a button. Finally, create a new form that will be the switchboard menu, and drag macros onto this form.

To see how the macros are to be created, assume that the input forms and reports have been designed and saved beforehand. Click on the Macro option in the database window, and select new. This will bring up the design view for a macro. Move the cursor to the first field in the Action column. Use the pull-down menu in this field and select OpenForm. Move the cursor to the Form Name field in the Action Arguments window (bottom part of the screen). Use the pull-down menu to list the existing forms (Access automatically recognizes only objects of type *Form* when the OpenForm action is selected). Select the desired form. Note that, using the Data Mode field in the Action Arguments window, you can also control whether the user should be allowed to only add new records, or edit existing records as well. To resize the form in order to display it on the entire screen, go to the next row, and select Maximize from the pull-down menu. Save the macro (you may wish to provide it the same name that you want to have appear on the button in the switchboard menu). To test the macro, you can highlight it and click on the Run button. This should pull up the desired form, and display it over the full-screen.

You can create a similar macro for the Edit Courses button. When two different macros perform similar actions, you can copy the first one and modify it as necessary (highlight the macro name in the macro list, choose the Copy option from the Edit menu followed by the Paste option, and provide it a new name when prompted by the system). Open the new macro in design view, and specify the name of the form that is to be opened by this macro. Save the modified macro and Run it to test.

To create a macro to print a report, start a New macro. Select OpenReport in the action column. Select the report in the Report Name field in the Action Arguments part of the window. If the user wants to preview the report, select Print Preview in the View field (this is the default option provided by Access). Save the macro. If the user wants to send the report directly to the printer without previewing it, select the Print option in the View field.

To create a macro to close the application, select Close in the Actions column of a new macro, and save it.

Once the four macros have been created, the switchboard form can be designed. Open a new design window for a Form. Move the form on the screen such that the Database window is visible (Click on the Form header bar, and drag it to one side if necessary. Similarly drag the database window if desired). Click on the Macros button in the database window to display all the macros. Highlight each macro and drag and drop it on the desired part of the form (one at a time). A button will be created that will have the same label as the name of the macro. You can edit this name by clicking on the button (*in form design view*) and retyping the text. You can also resize and move the button as desired. Additional labels (e.g. the application name) can be added to the form using the Label button in the Toolbox. Name this form and save it.

Once the form is saved, you can Open the form and it will show up on the screen. By clicking on each button, you can initiate the actions for the corresponding macros.

Once a particular action has been performed (e.g. a data entry form has been opened, and the user has made the desired changes to a table), you may wish to allow the user to return to the switchboard menu. This may be accomplished by adding to the data entry form a macro that closes the active window (you could use the one for the Close Employees Database button, or create a new macro). Open the data entry form in design view, and highlight and drag the macro from the database window onto the data entry form. Edit the label as necessary.

Starting an Application with an Autoexec Macro

A switchboard form itself can be opened by running a macro. Create a new macro that performs the action OpenForm with the switchboard form as the form name (in the Action Arguments window). Sometimes, we want to set up a system such that the switchboard form opens up whenever a user starts the database (i.e. when the user selects a database from the Open Database option in the File menu). This is done by creating a macro and naming it

Autoexec. Select the OpenForm action to open the switchboard form in the Autoexec macro. When a database is opened, Access checks for such a macro, and if it exists, it is executed right away.
Note that when this macro is included in a database, the database window does not automatically show up on the window. To view the database window, click on the Database Window icon in the Toolbar, or hit F11.

You can design applications where an user can click on buttons to move from one switchboard form to some other switchboard form. Each button must correspond to a macro that opens the desired form.