

RELATIONAL DATABASE TRAINING

RBase System V

Section 1.

PLANNING A DATABASE

1.1 SECTION OBJECTIVES

1. Participants will display a basic understanding of database planning by actively participating in a discussion of the importance of pre-design preparation.
 2. Participants will demonstrate an understanding of data categories, requirements and output by recognizing the three areas in a sample planning scenario.
 3. Participants will demonstrate their recognition of the planning steps by correlating them to three RBase Express Functions, Reports, Tables and Forms.
-

Planning a Database

1.2 PRE-PLANNING

When all is said and done, you want a database to take pieces of data and process them in such a way that it provides you with useful information. The type of information you want to generate will greatly influence the design of your database. The desired information will dictate the systematic design of the database. However, before the design of a database ever begins on the computer, the designer must sit down with pencil and paper and seriously consider the database user's need for information. He must be sure that the database is designed to meet the user's expectations, be they planners, administrators or researchers. He must also make sure that the database does not require data which exceeds the data collection capabilities of the users. The design of the database should enable the computer to transform individual bits of data into useful information which can then be presented in an easy-to-read format. Before you can design a database to do that, you must know what your target information is.

Planning a Database

The designer must conceptualize a consistently accurate tool which efficiently stores available data. In other words, the designer must make it easy for the user to store and retrieve data systematically. The database must be flexible enough to allow manipulation of data while, at the same time, protecting the integrity of that data. How does one begin to conceptualize such a tool?

1.3 The following questions will help you define the variables involved in the design of a database.

-
1. Who is using the information the database supplies?
 2. What are the goals and objectives of the database users?
 3. What tasks have they set before themselves?
 4. What categories of information is the user interested in; Stock Taking, i.e. inputs and outputs of the education system; Diagnosis, i.e. access and efficiency, utilization of resources, quality of education; Forecasting, i.e. size of student population, teacher requirements et cetera?
-

Planning A Database

1.4 Once you have pinpointed the information you want the database to provide, each step involved in building the database should be a step toward providing that information. As the construction of your database continues, keep in mind this very important question.

WHAT INFORMATION WILL APPEAR IN RBASE REPORTS ?

1.5 With a firm idea of the type and scope of information you want your database to generate you must now evaluate the data requirements of your database. In order to generate the information you want, you must be able to provide needed data. Again, the type of data you need will be dictated by the goals and objectives of the database user. Keeping in mind the information you want the database to provide, what data are important to your database?

Planning a Database

The following questions will help you focus on your data needs.

1. What categories of data are needed in order to generate the information you want the database to provide; pupil data; teacher data; school data?
 2. How will you arrange the data within the database?
 3. How can you standardize the data within the database?
 4. How can you link data through logical relationships?
 5. Will the database design allow the database to generate the desired information ?
-

Planning a Database

1.6 Understanding your data needs is vital to the systematic design of the database. Make sure you have a clear idea of the data you want to enter into your database before you begin building it. Take the time to outline your data requirements on paper before you begin work at the computer. Keep in mind the following question as you begin compiling your data.

WHAT DATA WILL BE STORED IN R:BASE TABLES ?

Planning a Database

1.7 The next step in planning your Database is determining how you will enter information into the database. **R:Base** gives you a variety of options when performing this function. This step is again directly related to the first two components of database planning and design. The following questions will help you plan for your data entry needs.

1. How is the data collected ?
 2. Are the tables you have designed prepared to store the data in its present state ?
 3. Must the data be manipulated in any way before it is entered into the database ?
 4. Look at your data categories, are you doing work that the computer will do for you? Can you make any changes in the way the data is handled ?
 5. Are you entering data from individual data collection sheets ?
 6. Are you importing data from another computer software program ?
-

Planning a Database

Deciding your strategy for data entry is especially important when you are dealing with large amounts of standardized data. Data can be entered into the database in a variety of ways. The type of data that we will deal with in this handbook is best entered in through use of a data entry form or a data import command. Both functions allow you to enter large amounts of data quickly and easily. The information found on data entry forms must be well coordinated with the first two steps mentioned above. As you formulate your database design, keep this question in mind.

DOES THE DATA ENTRY FORM MATCH THE
DATA REQUIREMENTS OF THE DATABASE ?

WHAT ARE REPORTS, TABLES , FORMS AND APPLICATIONS ?

1.8. REPORTS

Data that you retrieve and display for consideration is known as output. Output is presented in the form of a report. The report function in **R:Base System V** allows you to present data in a clear, concise manner. Once data has been entered into the database, **RBase** allows you to retrieve it in a variety of ways. **Reports Express** is the function within **R:Base** which allows you to design individual reports which can be stored in the memory of the computer for repeated use. The reports function enables you to present only the information you need when you need it. It is for this reason that planning ahead is so important. Once you have a clear idea of the information you need to make planning decisions, you will then be able to design reports that provide you with that information. **Reports Express** can present data taken directly from the database tables or it can display newly generated data, e.g. sorted information, totals etc. It can handle numerical as well as textual information. You will design reports to fit your database needs. Reports draw data from the database tables and it is therefore very important that you design your tables not only with data input in mind but data output as well.

Planning a Database

1.9. TABLES

A table is where the data within a database is stored. Tables are made up of columns and rows. Each column is defined by the type of data it contains. Each row contains a new data entry in a table. The **Express** function in **R:Base System V** makes creating tables very simple. When defining a table you only need to define the columns, not the rows. Each Database you create with **R:Base** can have up to 80 different tables. Each table can contain up to 800 columns. The structure of the tables depend on your needs. You will simply name each column and identify the type of data to be entered. **R:Base** prompts you with the mechanics but you must have a clear idea of the database structure before you can begin building tables.

You are most likely familiar with different types of data collection forms. Each form has a series of tables which contain data. The tables in **R:Base** will differ from those tables in many ways. We will explore why later on in the handbook. It is a good idea to begin thinking about database tables separate and apart from data collection tables.

Planning a Database

1.10 FORMS

Forms are used to input data into tables. They are helpful when you need to input the same type of data over and over, e.g. number of students in a school, employee contract numbers etc. **Forms Express** asks you to design a form just as you would on paper. You want to make sure that there is a place to enter each piece of data that the database structure requires. In order to load the table with the correct data, each column in a table must have a corresponding data entry field on a form. The forms are stored in the computer's memory and can be called up every time you want to enter in data. As you begin planning your database, remember that the computer is capable of generating information. Forms do not need to contain space for output, only input.

Planning a Database

1.11 APPLICATIONS

An important consideration in planning the design of a database is the database user. The database structure will provide the end user with the information he needs to make planning decisions but he is not the only one who will interact with the computer program. You must also consider the person who will manipulate the database on a daily basis.

As the designer of the database, you may want to limit access to data or allow only certain types of data to be entered. **R:Base** makes this possible through its **Application Express** mode.

There are some very practical considerations involved in the design of a database. For example, who is going to input the often huge amounts of data? It is unlikely that planners and decision makers will be spending hours inputting data. They, however, certainly will be concerned with the integrity of that data.

The person entering data may be unfamiliar with some of the more advanced program functions of **R:Base** and, as a designer, you want to make his job as simple as possible.

Planning a Database

Application Express enables you to design a series of menus that will help the new database operator do his job quickly and efficiently. Instead of learning **R:Base** commands, the operator simply selects options from a menu that you have designed. By designing an application you may simplify the operators job as well as protect your data from error.

Planning a Database

1.12 DATABASE PLANNING EXERCISES

1.12.1 Discussion Points-----

1. A database designer should first identify the type of information he wants the database to generate. Why?
2. Why is it important to design a database with the end user's goals and objectives in mind ?
3. How does the data stored in Tables relate to the information found in Reports ?
4. How is data entered into Tables ?
5. What is the major difference between Reports and Forms ?
6. Do Forms need to be exact copies of Reports in order to produce the output you want ? Why or Why not ?
7. Why does building an Application help protect your data ?

Planning Exercises

1.12.2 Analyzing a Database Problem Set

This exercise is best done in small groups. Using your knowledge of planning and databases, you will discuss the different steps in planning a database structure.

The Ministry of Education in country X wants to build a database which will be used in planning. Your job concerns reporting reliable statistical information to planners. You have been asked to design a database that compiles data taken from the Ministry's November Census of Schools. Using the questions found in this section as a guide, identify the different steps you would follow before you began constructing the database on the computer. Keep your answers general at this point, we will go into detail at a later point.

Planning Exercises

Once you have outlined your database needs, relate the different pieces of information to **R:Base Express Functions**. In other words, answer the three main questions found in section 1.

1. WHAT INFORMATION WILL APPEAR IN R:BASE REPORTS ?

2. WHAT DATA WILL BE STORED IN R:BASE TABLES ?

3. DOES THE DATA ENTRY FORM MATCH THE DATA
REQUIREMENTS OF THE DATABASE

Section 2.

A SAMPLE DATABASE

2.1 SECTION OBJECTIVES-----

1. Participants will become familiar with the basic structure of a database constructed with **R:Base System V** by attending a demonstration of a simplified version of a complete database.
2. Participants will become familiar with the **Express Functions of R:Base System V** by actively participating in an on-going discussion of the sample database during the demonstration of that database.

SECTION 3

Using RBase System V MENUS

3.1 SECTION OBJECTIVES

1. Participants will demonstrate an understanding of the two main menus, **RBsystem** and **R:Base** by successfully entering both menus from **DOS**.
2. Participants will demonstrate an understanding of **R:Base Main Menu** selection by successfully moving from one menu choice to the other.

3.2 INSTRUCTIONS

On the next page, you will find instructions on how to enter **RBase System V**. The diagram you see represents the two main menu paths in the program. The distinction between the two is an important one. **RBSYSTEM** leads you into the **Express functions** of the package. **Express** allows you to build tables, forms and reports without complicated commands. The handbook will deal mainly in this menu category but will branch out as your knowledge of the system deepens.

Using R:Base Menus

R:Base offers you a command mode in which you must be familiar with a series of commands in order to work with the data. It also offers you a prompt by example menu which guides you through the operations. Towards the middle of the workbook we will begin working with the command mode in order to retrieve information and change the structure of the database.

The instructions resemble, as closely as possible, the information you will see on the screen. Screen displays will be symbolized by asterix [*] above and below displayed information.

Because you want to save all the work you do on your own disk, you will be using a path command. This means that everything you do while working with **RBase System V** will be saved on your own disk. Simply follow the example and you should have no problems. It's very easy to make a typing mistake and suddenly find yourself somewhere in the system you do not want to be. Don't get discouraged -- there is always a way back.

Using R:Base Menu

3.3 ENTERING THE SYSTEM

Step 1. Turn on Screen and Computer

Step 2. When C> appears insert your disk into drive A

Step 3. Type in the commands that you see to the right
of the greater-than sign [>]

C:\ >A:

A:\ >Path c:\rbase

A:\ >Rbssystem

Using R:Base Menus

3.5 THE RBASE MENU

Instead of typing in **Rbsystem** at the **A:\>** prompt, type in **Rbase** as in the following example.

```
*****
```

```
A:\ >Rbase
```

```
*****
```

The **Rbase** prompt while in **DOS** will give you direct access to the **RBase Main Menu**. You can then choose from a four option menu.

We will practice moving from one menu to the other using the **Rbase** menu as an example.

```
*****
```

```
                R:BASE System V  
Copyright (C) 1983,1984,1985,1986 by Microrim, Inc. (Ver. 1.10 PC-DOS)
```

```
-----R:BASE Main Menu-----
```

- (1) R:BASE command mode
- (2) HELP for using R:BASE
- (3) Prompt By Example
- (4) Exit

```
*****
```

Using R:Base Menus

3.5.1 R:Base Command Mode

The first menu option on your screen is the **R:Base Command mode**. This menu choice allows you to use a wide range of **R:Base** commands. Choose the first option by pressing the return key. You will notice that a **R>** has appeared under the lower left hand corner of the menu.

R:BASE System V
Copyright (C) 1983,1984,1985,1986 by Microrim, Inc. (Ver. 1.10 PC-DOS)

R:BASE Main Menu (1) R:BASE command mode (2) HELP for using R:BASE (3) Prompt By Example (4) Exit

R>

In order to use command mode you must be familiar with the different **R:Base** commands. We will return to this menu option later in the handbook.

Return to the R:Base Main Menu by pressing Escape.

Using R:Base Menus

3.5.2 Help For Using R:Base

The second menu choice displays a help screen. When you are at the main RBase menu and need help with one of the program's tasks, you can call up this screen to receive specific help with any database management tasks. Choose the second option on the R:Base Main Menu to view the help screen.

HELP

Help is available for these database management tasks:

- 1 - Define a new database
- 2 - Modify an existing database structure
- 3 - Add rows to a table
- 4 - Look at table values, or information about table values
- 5 - Modify values in a table
- 6 - Create a new table from old tables
- 7 - Look at and modify R:BASE status values
- 8 - Manage drives, paths, files, and databases
- 9 - Build command files and R:BASE applications

At the H> prompt below, enter:

- A number (1 through 9) to display help for that task
- COMMANDS to list the R:BASE commands with help available
- An R:BASE command name to display help for that command
- ARGUMENTS to list commonly used syntax arguments
- SUPERMATH to list R:BASE functions

For additional HELP text, enter a command name. To leave HELP, enter END.

For the previous help menu press [ESC]

H>

Using R:Base Menus

As you become familiar with the program the different options on the Help menu will be more meaningful. For now, it is enough to know that it exists.

To exit the Help Screen type H>End

To return to the R:Base Main Menu press Escape

3.5.3 Prompt By Example

The third menu choice allows access to an extensive menu tree called prompt by example. In this mode, the screen will display messages prompting you to choose from a variety of menu choices. Choose the third option on the R:Base Main Menu to view the first prompt screen.

OPEN ...
OPEN opens an existing database. You must open a database to work with the data stored in it. If you do not want to open a database now, press [ESC].
Choose a database
TCAIRO STAFF EGYPT

Using R:Base Menus

The first message asks you to open a database. You would then simply move the highlighter to the database you wanted to open and then press return. As you learn the commands in R:Base Command mode you will not need to rely on prompts extensively but in the beginning it can be very helpful. We will not open a database now.

Press Escape to view the full Prompts menu.

Prompts are organized under these topics.
For a list of commands that have prompts, choose All commands.
To leave PROMPTS for R:BASE command mode, press [ESC].

Data Manipulation	Database Operations	Utilities
Look at data	Open a database	Application development
Print data	Exit	Database maintenance
Add data	Data Input	R:BASE environment
Edit data	Data Output	Edit an ASCII file
Import/export data	Create a database	DOS functions
Relational operations	Modify a database	All commands

As you can see, there are three columns of commands. We will not go through all the commands now but as your knowledge of R:Base deepens, you will be able to recognize the functions of the different commands.

To return to the R:Base Main Menu press Escape Twice

Using R:Base Menu

3.5.4 Exit

The last option exits the **R:Base** menu. Choose this option and return to the **RBsystem Menu** by typing **A:>rbsystem.**

SECTION 4

USING RBSYSTEM MENUS

4.1 SECTION OBJECTIVE

1. Participants will demonstrate an understanding of the **R:Base System V Main Menu** by successfully moving from one menu option to the other.
-

4.2 THE MAIN MENU

As you know, the **R:base System V** main menu offers you nine options. You returned to **DOS** to enter the **R:Base Main Menu**. The fifth option, **Rbase**, would have accomplished the same thing. It brings you into the **R:Base** menu without entering **DOS**.

Each option offers its own set of menus. The first choice, **Definition Express**, allows you to begin building a database. **Application Express** allows you to build your own menus. **Forms Express** allows you to build a data entry form.

Using RSystem Menus

Reports Express allows you to build a report. **RBEDIT** gives you direct access to files in order to edit. **FileGateway** allows you to import data from another software program. **Codelock** is used for converting R:Base command files from ASCII to binary format.

We will concern ourselves first with the simpler of the menu options. As you progress through the handbook you will access a growing number of menus. The first stage of the handbook concerns itself exclusively with the **Express Functions**.

Begin by choosing option one, Definition Express.

Definition EXPRESS
1983,1984,1985,1986 by Microrim, Inc. (Ver. 1.10 PC-DOS)

----- Definition EXPRESS Main Menu -----
(1) Define a new database
(2) Modify an existing database definition
(3) DOS functions
(4) Exit from Definition EXPRESS

Using RSystem Menus

4.3 DEFINITION EXPRESS

The **Definition Express Main Menu** gives you four different choices:

- 4.3.1. The first choice, **Define a new database** allows you to create a new Database. Defining a new Database begins by simply naming a database.
- 4.3.2 The second choice, **Modify an existing database**, allows you either to make changes in a database structure or just view the existing structure.
- 4.3.3. The third choice, **DOS Functions**, gives you access to selected Dos commands.
- 4.3.4 The last choice, **Exit**, will bring you back to the **RBase System V Main Menu**.

4.4 In this section we will look at the different options available to you in the **Modify an Existing database definition menu**.

Using the RSystem Menu

Choose the second menu option on the Definition Express Menu.

The following screen should appear.

Choose a database
TCAIRO STAFF EGYPT

4.4.1 OPENING A DATABASE

You are asked to choose a database to open. Opening a database allows you to access a number of database management features. As the number of defined databases increases, it becomes more difficult to remember what database contains the data you want. It is, therefore, a good idea to write down what data is contained in each database.

Move the highlighter to TCairo and press the return key.

Using RSystem Menus

4.5 THE DATABASE DEFINITION MENU

Database Definition Menu	
(1)	Tables
(2)	Views
(3)	Passwords
(4)	Rules
(5)	Forms
(6)	Reports
(7)	Return to Definition EXPRESS Main Menu

The **Database Definition Menu** that appears on your screen gives you a choice of tables, views, passwords, rules, forms, reports or the option of returning to the **Definition Express Menu**.

4.5.1 The first choice, **Tables**, provides access to the **Tables** menu. The tables menu allows you to add new tables, edit existing tables, remove tables or return to the **Database Definition Menu**.

Using RSystem Menus

- 4.5.2 The second choice, **Views**, provides access to the **Views Menu**. A **View** is a table created from other existing tables within the database. For example, you would use a view if you wanted to gather data from two or more existing tables and then make a report based on that data.
- 4.5.3 The third choice, **Passwords**, allows you to name a password that protects your data. By doing this you are able limit the number of people who have access to the database. Only those people who know the password can enter the database. This is very important for maintaining the integrity of your data.
- 4.5.4 The fourth choice, **Rules**, allows you to regulate the type of data that is entered into the database structure. This helps prevent errors by database operators. Up to twenty rules can be applied to each table. If the wrong type of data is entered into the database, an error message will be displayed on the screen. As the designer of the database, you are able to define the messages to suit your needs.

Using RSystem Menus

4.5.5 The fifth choice, **Forms**, allows you to access the **Forms Menu**. Once inside the **Forms Menu** you can create and edit, list, copy or remove data entry forms.

4.5.6 The sixth choice, **Reports**, allows you to access the **Reports Menu**. You will have the same capabilities in the **Reports menu** as you had in the **Forms menu**.

4.6 The handbook exercises are designed so that you enter and work with each of the above menus. The following sections are designed to guide you through the database features one at a time. The concept behind the handbook design is to present you with one difficulty at a time. You can view each one of the above menu options by moving the cursor to your choice and then pressing the enter key. To leave any menu you enter, press the escape key.

This section is completed. Leave the system by working your way back to the **RBase System V Main Menu** .

Choose Return To Dos

SECTION 5

DATABASE TABLES

5.1 INTRODUCTION

This section marks the beginning of a series of walk through exercises. You will be asked to analyze a variety of data collection material, perform database management tasks such as build tables, forms and reports as well as begin the design of a small database.

Each section is designed to expose you to the options available to you as a designer of a database. Write down questions, suggestions or comments whenever they arise and bring them up with one of your instructors.

5.2 SECTION OBJECTIVES

1. The participants will gain an understanding of the **Tables Menu** by exploring the different menu options.

Database Tables

2. The participants will understand the basic differences between the structure of database tables and data collection tables demonstrated by their ability to analyze sample tables for data content.

 3. The participants will gain an understanding of data base structure demonstrated by their ability to divide the information found in a data collection table into three distinct database categories, tables, forms and reports.

 4. The participants will be proficient in the steps involved in building database tables demonstrated by their ability to name tables, choose appropriate data columns, name columns and define data types.
-

Database Tables

5.3 LOOKING AT TABLES

With the **Database Definition Menu** on your screen,
CHOOSE OPTION 1, TABLES

The Following Screen Should Appear

```
-----Tables Menu-----
(1) Add a new table to the database
(2) Change an existing table
(3) Remove a table from the database
(4) Return to the Database Definition Menu
```

By choosing "tables" you display the **Tables Menu**. You now have a choice of four options. The first option allows you to add a table to the database. The third option lets you remove a table from the database. For the moment, you will not be creating or removing tables so move the highlighter to the second option.

CHOOSE [2] CHANGE AN EXISTING TABLE.

Choosing this option does not necessarily mean that you are going to change anything. The option also allows you to look at the different tables without changing any of them.

Database Tables

5.4 Tables in Data Collection Forms

Look at Figure 1. on this page. The table is taken from an Egyptian Ministry of Education data collection form.

What information can be gathered from this table ?

Study the table for a moment and think about how the information is organized. Write down as many categories of information you can find within the table. For example, first grade boys etc.

Figure 1.

Number of Students			Number of Classes	Grade
Total	Girls	Boys		
				First
				Second
				Total
			Number of New Entrants	

Database Tables

Data categories found in data collection tables form the database tables. Each database table contains columns and rows. Each column is given a name and represents a category of information; for example, first grade boy. The rows represent individual entries. In other words, if you had two columns representing the first and last name of employees, every time you entered a new person into the table, it would form a row.

**If each category of information represents a column,
how many columns should you have in Figure 1 ?**

Database Tables

5.5 Now move the highlighter to **example1** and press the return key. The table named **example1** should appear on the computer screen.

Move the highlighted cursor through the table and notice how many columns there are.

* To move the highlighter first push the return key and it will move down to the table columns. You can move the cursor from one column to the other by pushing the **Tab** key. To move the cursor in the opposite direction, push the **number 4** key on the cursor control board key.

If you hit the wrong key and erase letters by mistake, retype the erased letters. Look at the tables in this handbook if you cannot remember the letters you erased. To avoid erasing letters, try not to touch any keys except the spacebar or number 4 key when moving through the table.

Database Tables

Below you will find table example1 as it appears in R:Base System V. Look at data collection table number 1. in the Pre-Primary Data Collection Form. What differences do you notice between the columns in the data collection table and the columns in the database tables ?

Example1

An asterisk (*) identifies key columns
A plus sign (+) identifies computed columns

#class1	Firstboy	Firstgir	Totfirst	#class2	seconboy	secongir
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER

totsecon	totclass	totboys	totgirls	total	bnewents	gnewents	Tnewents
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER

Database Tables

The first thing you should notice is that the format of the database table does not resemble the table found in the Ministry's data collection form. The columns in the data collection table are only part of the database tables you want to define. You must therefore translate the Ministry's form into a format that the **R:Base** package will accept and that serves your needs.

If you remember, we said that when building a database you must first ask yourself what information you want to appear in reports. The report will be the final product. It retrieves data it finds in the database tables. How you design your tables will influence the type and extent of the information that appears in reports. If you design your columns carefully, you will be able to produce reports that provide useful information. If not, you may find yourself with information of little use. For instance, if you choose to name one of your columns "girls", you will later find that the report will only give you a number of girls; you won't know what school they study in or what grade they are in. The information may be interesting but of little use in planning.

Database Tables

One important thing to remember is that columns in data collection tables are not the same as columns in database tables. Remembering the three basic parts of the database, Reports, Tables and Forms, let's look at how a table from a data collection form can be divided into the three areas of database management. As we go through the exercise, try to think about why the information is divided the way it is.

Database Tables

5.6 Data Collection Forms and Database Management Areas.

If we think of a database as a tool which is able to store existing data as well as generate new data, analyzing data collection tables is much easier. You must find the relationships between the structure of the data collection table and the database tables where the data will be stored.

The data collection table you looked at contains two types of data, input and output. Input is data you enter into the database while output, for example, would be considered the totals found at the bottom of the different columns. When designing a database structure, you must make a clear distinction between input and output. The database table columns accept input and reports display output. If you are inputting totals into database tables, you are most likely doing a job the computer would do for you.

Data collection tables can be divided into information categories which are associated with specific functions in **R:Base System V**. The important thing to remember is the distinction between input and output.

Database Tables

Look at Figure 2 on the next page.

The sample data collection table we looked at in Figure 1 has been divided into colour coded sections which represent three different data management areas, tables, forms and reports.

The yellow section of the table corresponds to the data entry Form design. It can be edited in any way without interfering with the database structure.

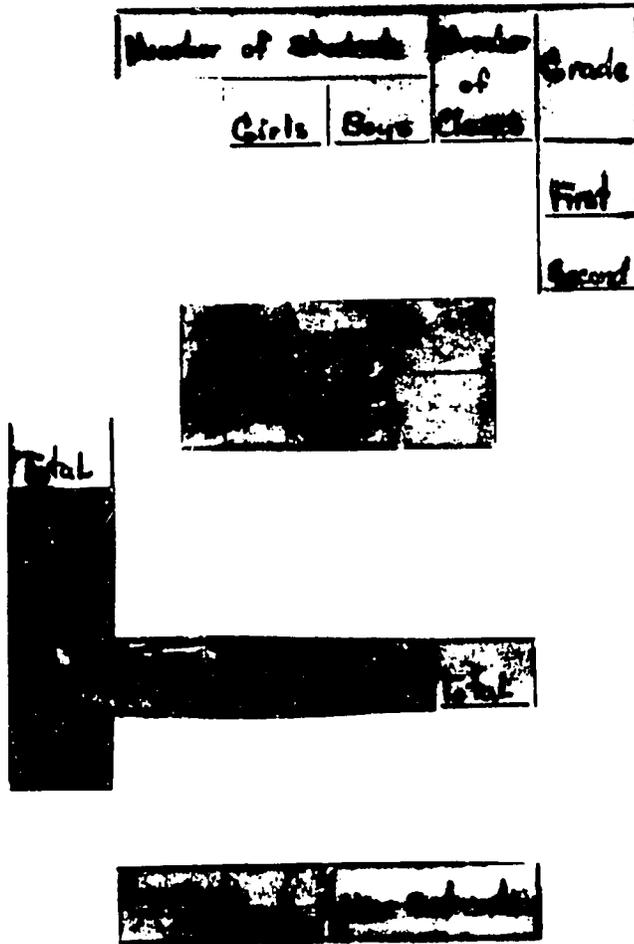
The data columns, in blue, correspond to the columns in the database table and are meant to hold input data.

The report variables, in green, correspond to database output. They contain data that R:Base will compute for you.

Note that the data found in the blue sections is entered into the database as input and that the data found in the green sections need only appear in reports.

Database Tables

Figure 2.



Form Design

Table Columns
{ Input }

Report: [Illegible]
[Illegible]

5.7 CREATING DATABASE TABLES

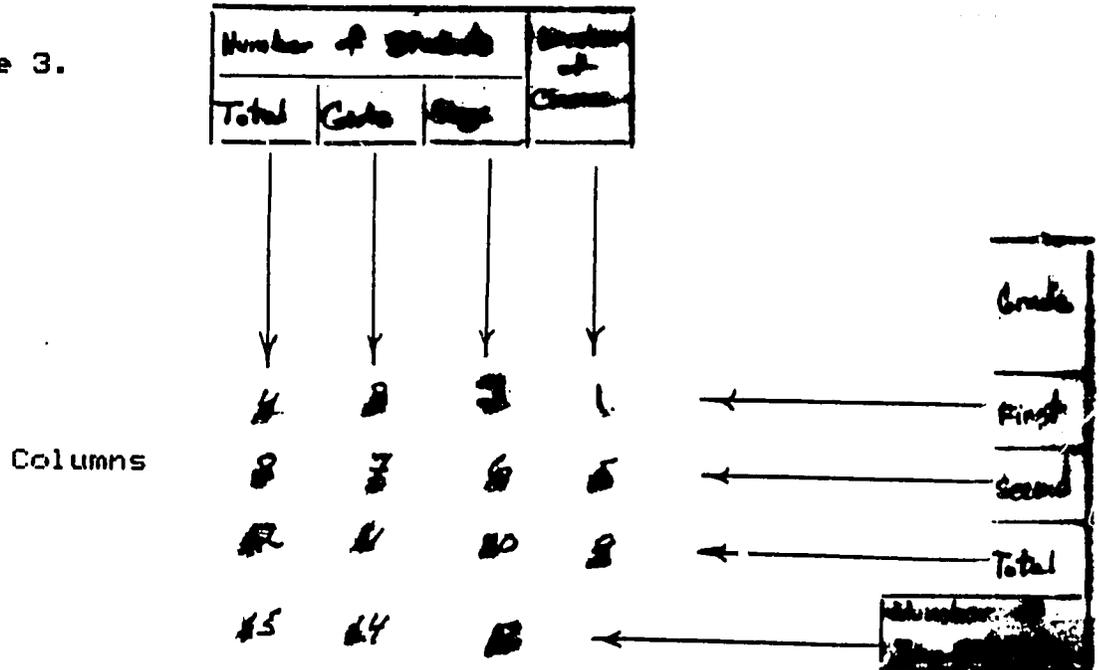
R:Base makes creating tables very simple. Once you have determined the number of columns you need to meet your data requirements, you can begin the process of defining tables. Let's look at how one determines how many columns are needed in a table.

Remember the categories of information you wrote down earlier while looking at Figure 1. Each one of those categories represents a column. Each column must be given a name which can contain up to, but not more than, eight [8] characters. When you looked at table example1 on your screen, you saw 15 columns. Figure 3 shows how the 15 columns were defined and named.

The numbers in blue represent the database columns. The information contained in these columns is cross-referenced by class, sex and grade. Doing this allows you to retrieve useful information later on when you are designing reports.

Database Tables

Figure 3.



Column Name	Column Number	Matrix Operation	Information Category
#class1	= 1	First + # of Classes	# of Classes in First Grade
Firstboy	= 2	First + Boys	# of Boys in First Grade
Firstgir	= 3	First + Girls	# of Girls in First Grade
etc.....			

A complete list of column names follows.

Database Tables

Figure 4.

Terminology Used in Naming Columns

Table : Example1

<u>Column Name</u>	<u>Cross Referenced Column Data</u>
#class1	< Number of First Grade Classes
Firstboy	< Number of First Grade Boys
Firstgir	< Number of First Grade Girls
Totfirst	< Total Students in First Grade
#class2	< Number of Second grade Classes
Seconboy	< Number of Second grade boys
Secongir	< Number of Second grade girls
Totsecon	< Total students in Second Grade
Totclass	< Total number of classes
Totboys	< Total number of boys
Totgirls	< Total number of girls
Total	< Total number of Students
bnewents	< Number of boy new entrants
gnewents	< Number of girl new entrants
Tnewents	< Total of new entrants

Database Tables

5.8 COMPARING DEFINED TABLES

Look at Table Example2 on your screen.

To move to table Example2 simply push the **Escape** key and you will be brought back to the tables menu.

Move the cursor to option 2 and press return.

Move the cursor to Example2 and press return.

Answer the following questions as you compare the two tables. Copies of the tables are found on the next page.

-
1. What is the difference between tables Example1 and Example2 ?
 2. Which Table has more columns?
 3. Why?
-

Database Tables

Figure 5.

Tables: Example1 and Example2

Example2 An asterisk (*) identifies key columns.
A plus sign (+) identifies computed columns

#class1	firstboy	firstgir	#class2	seconboy	secongir	bnewents	gnewents
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER

Example1 An asterisk (*) identifies key columns.
A plus sign (+) identifies computed columns

#class1	Firstboy	Firstgir	Totfirst	#class2	seconboy	secongir
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER

totsecon	totclass	totboys	totgirls	total	bnewents	gnewents	Tnewentf
INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER

Database Tables

Figure 6.-----

Terminology Used in Naming Columns

Table : Example2

<u>Column Name</u>	<u>Cross Referenced Column Data</u>
#class1	< Number of First Grade Classes
Firstboy	< Number of First Grade Boys
Firstgir	< Number of First Grade Girls
#class2	< Number of Second grade Classes
Seconboy	< Number of Second grade boys
Secongir	< Number of Second grade girls
Bnewents	< Number of boy new entrants
Gnewents	< Number of girl new entrants

Database Tables

As you compare the columns in tables **Example1** and **Example2**, you should notice that **Example2** contains far fewer columns due to the fact that it does not contain totals. In the first example, every box in the data collection form is represented by a database column. That is one possible way to build a database but it does not take full advantage of the mathematical capabilities of **R:Base System V**. By building your database in that fashion you must perform all the calculations before the data can be entered into the tables. **R:Base** enables you to generate new data; in other words, it will add up all the totals for you and display them in a report.

Database Tables

5.9 DATABASE TABLES EXERCISES

Quickly review Figure 2 and tables Example1 and Example2 keeping in mind the different data areas and how they relate to database management tasks.

Now look at Figure 7. below. This data collection table was taken from an Egyptian Ministry data collection form meant for use at the primary level.

Using Figure 7. as practice, the next section will show you how to create a table named Primstud.

Figure 7.

Number of Students			# of classes	Grade
Total	Girls	Boys		
				1
				2
				3
				4
				5
				6
				Total

Database Tables

5.9.1 PLANNING THE TABLE

- STEP 1.** Take a piece of paper and divide the table into three separate sections as was done in Figure 2. Review the Figure 2. to remind yourself of the 3 sections.
- STEP 2.** Remembering that tables are used to store input, determine which boxes in the table would contain input.
- STEP 3.** Define the columns and assign each one a name that contains no more than eight characters.
- STEP 4.** Double check your plan. Are you planning a table that is designed to hold input only ? Are you doing work the computer will do for you ?
- STEP 5.** How many columns have you defined ? Check with the group and compare results. What changes, if any, would you make in the data collection form ?

Database Tables

5.9.2 CREATING THE TABLE

STEP 1. Return to the tables Menu.

STEP 2. Choose the first menu option, **Add a new table to the database.**

STEP 3. The following screen should appear.

```
*****
Enter the name for this table


```


```
*****
```

STEP 4. You are asked to supply the table with a name. The name can only contain 8 characters and should remind you of the data the table will contain. In this exercise, name the table **Primstud**, short for Primary Students. Once you have named the table, press the return key to enter it.

Database Tables

STEP 5. The cursor will move down to the first column space. Type in the name of your first column and press the return key. [The return key is always used to execute an action unless otherwise specified.]

STEP 6. The following screen will appear.

Enter or change the column names

An asterisk (*) identifies key columns
A plus sign (+) identifies computed columns

Choose column data type

TEXT CURRENCY INTEGER REAL DOUBLE DATE TIME NOTE COMPUTED

This is a data type menu. For each column you name, you must define the type of data that will be entered into it.

You have a choice of nine data types. For this example we will deal with integers. In examples to come, we will use other data types and learn the steps involved in defining the amount of space each column is allocated. When you choose integer, **RtBase** automatically allocates column space.

Choose **Integer** for each column data type.

Database Tables

STEP 7. You are asked if you want the column to be a key column. A key column should contain unique data that is referred to later when you want to retrieve data. You would designate a column to be a key if it contains unique data such as school identification codes; in other words, data that is not duplicated within the table. Defining key columns helps R:Base sort through data faster. For now, answer no to the question. You can always make a column a key later when you have learned more about the database structure.

STEP 8. Continue this process until you have typed in all your column names. You have just completed the first step in building your database.

STEP 9. Exit from the table by pressing the **Escape** key.

SECTION 6

DISCUSSION POINTS

PRE-DESIGN AND CREATING TABLES

6.1 TYING YOUR DATA TABLES TOGETHER-----

As the number of tables in your database grows, you have access to more and more data. Storing data is one use of **R:Base System V** but the features of **R:Base** go far beyond the simple storing of data. In order to take full advantage of your new relational database, you must plan your tables with the whole picture in mind.

On the next page, Figure 5 shows the data categories contained in the Egyptian pre-primary data collection tables. As it appears now, each table stands separate and apart. In order to be able to relate data in one table to data in another table, you must find a way to link all the data contained on this sheet together.

Discussion Points

6.2 DISCUSSION QUESTIONS

Discuss the following questions in groups and come up with some possible solutions.

1. For whom do you think the data contained in the tables was collected ? Administrators, planners?
2. Why did the designers choose these data categories ? Are the tables collecting research data, planning data, stocktaking data, flow data ? Does it make a difference ? Explain your answer .
3. Do the tables have anything in common? How would you group the tables into data management categories, e.g. research, diagnosis, forecasting and planning ?

Discussion Points

5. Is there any reason to link the tables together ? Explain your response.
 6. Can you suggest changes in the data collection forms ?
 7. What is the difference between transferring the information you find in Figure 5 into a database and designing a relational database based on Figure 5's data categories ?
 8. How can pre-design planning help you define a more powerful and efficient database ?
-

Pre-Design and Creating Tables

Figure 5

Table 1

Classes
first grade girls
first grade boys
second grade girls
second grade boys
new entrants

Table 3

Muslim Boys
Muslim Girls
Christian Boys
Christian Girls
other Boys
other Girls

Table 4

Egyptian Boys
Egyptian Girls
Sudanese Boys
Sudanese Girls
Palestinian Boys
Palestinian Girls
Arab boys
Arab girls
Non-Arab boys
Non-Arab girls

Table 5

Table 7

Boarding Boys
Boarding Girls

Table 8

Male Principals
Female Principals
Male supervisory Teachers
Female Supervisory Teachers
Other Male Employees
Other Female Employees
Male Service Employees
Female Service Employees

Table 17

First Grade Tuition
First Grade Board
First Grade Transportation
First Grade Fees
Second Grade Tuition
Second Grade Transportation
Second Grade Board
Second Grade Fees

SECTION 7

FORMS EXPRESS

7.1 SECTION OBJECTIVES

1. The participants will demonstrate a familiarity with the function of Data Input Forms and how they relate to Database Tables by actively participating in a discussion of Forms pre-design.
2. The participants will gain an understanding of the **Forms Menu** by exploring the different menu options.
3. The participants will understand the basic design components in **Forms Express** demonstrated by their ability to divide a sample form into two sections, data input fields and format information.
4. The participants will gain a basic proficiency in building data input forms demonstrated by their ability to define a form design, locate data input fields and relate those fields to table columns while building a sample form.

Forms Express

7.2 Database tables store data that has been loaded into the computer. You have built a database table; your next task is to decide on a method of loading data into that table. Building data entry forms in Forms Express is one way to enter data and will be the focus of this section.

There are two major parts of building a form. The first part is designing the lay out of the form. In Figure 6, the pre-primary data collection table we used in the previous chapter is divided into two sections. The first section is information that you can edit freely and position anywhere on the screen. It makes up the basic form design and is the area outlined in yellow.

Once you have decided on a form design, you must locate the fields where data can be entered on the forms. You must locate as many fields as you have columns in the corresponding table. Each field corresponds to a column name and data will be entered into that column through that field. The blue area outlined in Figure 6 represents columns and, therefore, data entry fields.

Forms Express

7.3 LOOKING AT FORMS

STEP 1. Begin at the R:Base System V Main Menu.

Move the cursor down to **Forms Express** and push the enter key. The **Forms Express Menu** will appear.

Forms EXPRESS
Copyright (C) 1983,1984,1985,1986 by Microrim, Inc. (Ver. 1.10 PC-DOS)

Forms EXPRESS Main Menu	
(1)	Choose a database
(2)	List directory
(3)	Change directories
(4)	Exit from Forms EXPRESS

Forms Express

STEP 2. The cursor is on option number 1, **Choose a Database.** Press the return key. The **Database menu** will appear.

Choose a database
TCAIRO STAFF EGYPT

STEP 3. Choose the TCairo database. The **Form Options menu** will appear.

انتخب	Form Options
(1)	Edit/Create a form
(2)	List forms
(3)	Copy a form
(4)	Remove a form
(5)	Return to Forms EXPRESS Main Menu

Forms Express

STEP 4. The **Form Options** menu gives you a choice of five options. **Edit/Create a form** will allow you to edit an existing form or create a new one. **List a form** will display a list of forms and the tables they are based on. **Copy a form** will make a copy of a form and allow you to rename the copy, and the fifth option puts you back in the **Forms Express** menu. **Choose option 1, Edit/Create a Form,** and the following menu will appear.

Choose a form
example2 (New)

Forms Express

STEP 5. Choose **Example2** and the **Form Definition** menu will appear. Below the menu you see **Form Example2.**

Form Definition Menu			
Edit	Expression	Customize	Draw

Database: Tcairo
Form: Example2

Number of Students		Number of Classes in School		Grade Level
Girls	Boys			
S	E	S	E	First
S	E	S	E	Second
S	E	S	E	Number of New Entrants

[ESC] Return [F3] Review [F7] Prev table [F8] Next table [F10] Help
Form: example2 Table: Example2

Forms Express

7.4 The Form Definition Menu at the top of the screen lets you make various changes in the form design and the data entry fields.

Edit- allows you to move freely throughout the form. Use it to build forms or make changes in existing forms. While moving through the form, be careful to use the cursor control keys only. If you use the spacebar or back space key you will erase parts of the form.

EXPRESSION- allows you to define expressions. As we are building a simple form we will not use this option right now.

CUSTOMIZE- allows you to change default settings in order to better meet your design needs.

DRAW- allows you draw lines to create boxes and outlines.

Choose Edit to enter and move around in Form

Example2

7.5 DATA ENTRY FIELDS AND THE FORM DESIGN

The form you are looking at was created to enter data into table **Example2**. The copy of the form that appears in the handbook above is divided into two coloured areas. Again, the area outlined in yellow can be edited freely. The S and E that you see in each blue box is the data entry field. When you are ready to load data, the cursor appears between the S and the E. **The S means Start and the E means End.** They mark the length of the data entry. The default length for integers is 8 spaces.

Every time you enter data in-between an S and an E, it is programmed to form a row in the corresponding column in the database table.

7.6 PLANNING A FORM

Look at Figure 7. In the chapter on Database Tables, you built a table based on this data collection table and named it **Primstud**. You will now build a data entry form that will allow you to load **Primstud** with data.

Forms Express

Figure 7.

Number of Students			# of classes	Grade
Total	Girls	Boys		
				1
				2
				3
				4
				5
				6
				Total

Forms Express

7.7 CREATING A DATA ENTRY FORM

- STEP 1.** Leave **Form Example2** by pressing **Escape** twice. Normally when you finish building a form, the **Save Changes** option would be chosen. The **Return** option would put you back in the **Form Definition Menu**. In case you erased something in **Example2** by mistake, choose **Discard Changes**.
- STEP 2.** Take a piece of paper and draw your form design before you begin work on the computer. Make sure you leave enough room for data entry fields. Refer to **Step 5**.
- STEP 3.** Now go back to the **Forms Option** menu and choose **Edit/Create a form**. Now choose **(New)**. You are asked to name your new form. You are building a form to enter data into your **Primstud** table. To make it easy to remember what table goes with each form, name your form **Primstud**.

Forms Express

STEP 4. Express will ask you if you want to customize form characteristics. To look at the default settings, answer yes. The following screen will appear.

Number of Students		Number of Classes in School	Grade Level	
Girls	Boys			
S	E	S	E	First
S	E	S	E	Second
S	E	S	E	Number of New Entrants

[ESC] Return [F3] Review [F7] Prev table [F8] Next table [F10] Help
Form: example2 Table: Example2

Press [ESC] when done

Form Characteristics

Assign passwords for this form? [N/A]
 Read-only password: _____ Modify password: _____
 Clear the screen before form use? [Yes]
 Clear the screen after form use? [Yes]
 Display a status line during form use? [Yes]
 Do you want custom colors for the form? [No]
 Foreground color: _____ Background color: _____
 (Press [ENTER] for a color palette)

Do you plan to use the form with the ENTER command? [Yes]
 Do you want to change the menu? [No]

Add Duplicate Edit again Discard Quit

Do you plan to use the form with the EDIT command? [Yes]
 Do you want to change the menu? [No]

Edit Save Add new Delete Reset Previous Next Quit

] Up [] Down
Form: x Customize form

Forms Express

The form characteristics default settings can be changed to add passwords, menu selections, colour backgrounds etc. As your knowledge of **Forms Express** increase, you can experiment with these settings. To make things simpler, we won't change any of the settings now. **Push Escape to continue.**

STEP 5. **Express** now asks you to choose a table.

Choosing a table means that you are matching your new form to a table, i.e. which table do you want to load with data using this form ? Choose **Priastud.**

Forms Express

STEP 6. Express will now ask if you want to customize the table characteristics. To look at the default settings answer yes. The following screen will appear.

```
*****
Press [ESC] when done
Table Characteristics

Do you want to add new rows to the table? ..... [Yes]
Do you want to replace existing rows in the table? ..... [Yes]
  Is the replace automatic when the user leaves the row? ..... [Yes]
Do you want to delete rows from the table? ..... [Yes]
  Restrict the delete to the current table? ..... [Yes]
Is this table on the MANY side of a ONE-to-MANY relationship? ..... [N/A]
Do you want to define a region? ..... [No ]
  Do you want a border around the region? ..... [N/A]
  How many lines in the border - enter 1 or 2: _
  Do you want custom colors for the region? ..... [N/A]
  Foreground color: _____ Background color: _____
  (Press [ENTER] for a color palette)

] Up  [ ] Down
Form: x      Customize table      Table: xx
```

```
*****

The default settings allows you to add,
replace or delete rows from the table. The
settings do not need to be changed so push
Escape to continue.
```

STEP 7. The **Form Definition Menu** will appear. Choose the **Edit** option from the menu and a blank screen will appear. You will not see any boxes on the screen. You must begin copying the form you drew on paper to the computer screen.

Forms Express

STEP 8. You are now ready to begin building your form. If you look at the bottom of the screen you will find a list of keys that will be of great use in designing forms.

The F1 Key will insert a new row space into your form.

The F2 Key can be used to delete rows from the screen.

The F3 Key is especially helpful when you are locating data entry fields. If you forget the name of a column, push F3 and it will list all the column names in the table.

STEP 9. Move the cursor to the place on the screen where you want to begin and then type in the form headings you have written down on your paper. When you have finished with the form design, it is time to locate the data entry fields.

Forms Express

STEP 10 Move the cursor to the space where you want to locate the first data entry field.

Locating data entry fields should be done in the order you want to load the data. When you are ready to load data with the form, the cursor will move from one data entry field to the other in the exact order that you created them. Before you begin locating fields, look at the example below.

In Form **example2**, the fields were located in the following order.

Form Definition Menu			
Edit	Expression	Customize	Draw

Database: Tcairo
Form: Example2

Number of Students		Number of Classes in School	Grade Level	
Girls	Boys			
S	E	S	E	First
S	E	S	E	Second
S	E	S	E	Number of New Entrants

[ESC] Return [F3] Review [F7] Prev table [F8] Next table [F10] Help
Form: example2 Table: Example2

Forms Express

- STEP 11** Using the form design you have drawn on paper, number the data entry fields in the order you want to create them.
- STEP 12** Move the cursor to exactly where you want to locate the first field. In order to locate a field push F6. Express will ask you to enter the column name you want this field to match. Type the exact column name or the screen will flash an error message. If you forget the name of the column, push F3 and the list of column names will appear on the screen. If you make a mistake, the backspace key can be used to correct it. Press return after you have typed the correct column name.
- STEP 13** Answer **No** to the questions, "Do you want to define an Expression and Do you want to define field characteristics ?"

Forms Express

STEP 14 Now move the cursor to the exact place you want to locate the start of the entry field and type S. Type E where you want the field to end. Continue the process until you have located all the data entry fields. Remember to leave enough room between the headings for the data entry fields. If there is not enough room, the data entry field will erase your heading and you will have to type it again. Push Escape to leave the form. Choose **Save Changes**.

7.8 EDITING THE DATA ENTRY FIELDS

If you want to move or erase a data entry field, push the shift and F2 key at the same time and the field will be erased. Pushing F2 alone will only erase the S and E, it will not erase the field. You can use the space bar to erase headings. It will also erase the S and E but the field will remain in the computer memory. If you erase an S or E with the spacebar by mistake, you can just retype them.

Forms Express

7.9 Discussion Points

Discuss these points with trainers and fellow participants.

1. Is your form easy to read ? How could you improve it?
2. Is there a data entry field for every column in table **Prinstud** ?
3. Does your form include headings to make it easily recognizable to a database user ?

SECTION 8

USING REPORTS EXPRESS

8.1 SECTION OBJECTIVES

1. The participants will be familiar with the concept of generating informative and well presented ad hoc reports.
 2. The participants will demonstrate an understanding of ad hoc report design by accurately designing sample reports on paper.
 3. The participants will be proficient in the manipulation of **Reports Express Menus** demonstrated by their ability to view all menu options.
 4. The participants will be proficient in the creation of reports demonstrated by their ability to produce a **Report** design in **R:Base**.
-

Reports Express

8.2 WHY GENERATE A REPORT ?

One of the best features of **Ribase** is its capability to generate reports. As more and more data is entered into your database, you will not want to look through pages and pages of reports to find what you are looking for. **Reports Express** lets you design individual reports to suit your data needs.

You will be able to design reports with specific purposes in mind. For example, a sample table contains general school information, i.e. name, address and telephone number of the headmaster etc., as well as student enrollment data.

One person using the database may want a list of addresses and telephone numbers. A planner may only want to look at the student enrollment data. Using **Reports express**, you are able to create a report for both of them. When they want to print out a report, they will get only the information they are interested in .

Reports Express

8.3 LOOKING AT A REPORT

STEP 1. To look at a sample report, begin by choosing Reports Express from the main R:Base System V menu. The following menu will appear.

```
Reports EXPRESS Main Menu
(1) Choose a database
(2) List directory
(3) Change directories
(4) Exit from Reports EXPRESS
```

STEP 2. The Reports Express Main Menu gives you four options. Choose the first option, Choose a Database. The Choose a Database Menu will appear.

```
Choose a database
TCAIRD STAFF EGYPT
```

Choose Tcairo.

Reports Express

STEP 3. The **Report Options Menu** gives you five options. You should be familiar with the rhythm of the different menu choices by now.

Report Options Menu	
(1)	Edit/Create a report
(2)	List reports and associated tables
(3)	Copy a report
(4)	Remove a report
(5)	Return to Reports EXPRESS Main Menu

3.1 Edit/Create a Report- allows you to edit an existing report or create a new one.

3.2 List reports and associated tables-will give you a list of existing reports and the tables from which they retrieve data.

3.3 Copy a Report-allows you to make a copy of an existing report. You are asked to give the name of the report you want to copy and then assign it a new name. You will then have two copies of the same report but they will have different names. As a general rule in RBase, no two tables, forms or reports can have the same name.

Reports Express

3.4 Remove a report—allows you to remove a report from the database. Once you have removed a report you can not get it back so you should be careful with this option.

3.5 Return to Express Main Menu—will put you back in the previous menu.

STEP 4. Choose option number 1, **Edit/Create a report**. The **Choose a report** menu will appear.

Classes (New)	Choose a report
---------------	-----------------

Reports Express

STEP 5. There is only one report, **Classes** ,in the menu. Choose it. The **Reports Definition Menu** will appear. Below it you will find the **Classes** report.

Reports Definition Menu			
Edit	Expression	Configure	Draw

RH
RH
RH
RH
PH
PH
PH
D
D
D
RF
RF
RF
RF

Report on Classes and Students
Database TCairo

Grade 2		Grade1		New Entrants					
Girls	Boys	Girls	Boys	Girls	Boys				
S	E	S	E	S	E	S	E	S	E
S	E	S	E	S	E	S	E	S	E
									TOTALS

STEP 6. The **Reports Definition Menu** gives you four options. In this simple report, you need to concern yourself mainly with the **Edit** option. Just as in **Forms Express**, you should choose **Edit** to enter the report and move around the screen.

Reports Express

8.4 THE REPORT SCREEN

Building a report is considerably more complicated than building a form. The first report you look at is very simple. The form you build will also be very simple. This section will describe what you see on the report screen.

8.4.1 In this report there are four main sections. Look at the left hand side of the screen and you will see a column containing the letters RH, PH, D and RF. These letters stand for Report Header, Page Header, Detail and Report Footer.

8.4.2 **REPORT HEADER**- contains descriptive information that will appear on the first page of the report only.

Reports Express

- 8.4.3 PAGE HEADER-** contains information which will appear at the top of each page. This report is designed so that Pre-Primary Enrollment, Grade 2, Grade 1 and New Entrants will appear at the top, or head, of any data that is printed. You can think of the page headings as titles of data columns in the report.
- 8.4.4 DETAIL-** contains the rows of data that will be displayed in the report. It may appear that the report only has one line but, when the report is printed, a line will be created for each row generated. It is in this section that you will locate data fields just as you did in **Forms Express**. In this case, however, they will not represent data entry fields, they will represent data display fields.
- 8.4.5 REPORT FOOTER-**contains the totals of the columns displayed. The totals will appear at the bottom of the report.

Reports Express

8.5. SCREEN SIZE

In **Forms Express**, the size of your form could not be bigger than the screen. In **Reports Express**, however, you can go beyond the screen if you need extra room. A screen can only display 78 columns at one time. **Reports Express** allows you to make a report as wide as 255 columns. If you move the cursor to the right it will bring you to an empty section where you could continue working. To come back to the report move the cursor to the left. If you create a report that is wider than one screen, you must print it on wide paper or in a compressed, meaning very small, print. Ask a trainer how to do this when you want to print a report.

Reports Express

8.6 THE FUNCTION KEYS

Function keys are used for many different reasons. **Reports Express** makes use of all the Function keys and they can sometimes be confusing. Before you begin building a report, read the following explanations very carefully.

Figure 8.1

Table 8.1 Report Creation Keyboard Functions	
Key	Function
[]	Moves the cursor one space to the right
[]	Moves the cursor one space to the left
[Ctrl][]	Moves the cursor to column 255
[Ctrl][]	Moves the cursor to column 1
[]	Moves the cursor up one line, from the first line of a section with EXPAND on, creates a new line above the current line
[]	Moves the cursor down one line, from the last line of a section with EXPAND on, creates a new line below the current line
[Ins]	Inserts a blank space at the cursor position
[Del]	Deletes the character at the cursor position
[Tab]	From the work area, moves the cursor to the next tab setting. From the mark area, moves the cursor to the work area.
[Shift-Tab]	From columns 2 through 255 of the work area, moves the cursor to the previous tab setting. From column 1 of the work area, moves the cursor to the mark area.
[Home]	With EXPAND off, cursor goes to the top left corner of the report. With EXPAND on, cursor goes to the top left corner of the current section
[End]	With EXPAND off, cursor goes to the bottom right corner of the report. With EXPAND on, cursor goes to the bottom right corner of the current section
[Enter]	Chooses the current menu option
[F1]	Inserts a blank line above the current line
[F2]	Deletes the current line
[F3]	Displays the variables and columns of the current driving table
[F4]	Toggles the repeat character function on and off
[F5]	Resets the value of the current field
[F6]	Locates a field
[F7]	With EXPAND off, moves the cursor to the previous section. With EXPAND on, moves the cursor to the previous section, creating one if it does not already exist
[F8]	With EXPAND off, moves the cursor to the next section. With EXPAND on, moves the cursor to the next section, creating one if it does not already exist
[F9]	Toggles the EXPAND mode on and off
[F10]	Displays a help screen
[Shift-F2]	Deletes the current field
[Esc]	Displays menu or exits editor

8.7 SPECIAL ATTENTION

Pay very close attention to the [F7], [F8] and [F9] keys. The [F9] key controls a function called **Expand**. This function does not expand the screens, it expands the sections mentioned above.

8.7.1 Look at the bottom of your screen. Do you see **Expand** written towards the center of the screen ?

8.7.2 Push the [F9] function key. Now do you see **Expand** written ?

8.7.3 Make sure the **Expand** function is on. It is on if **Expand** is written at the bottom of the screen.

8.7.3 Move the cursor down the screen. What happens when you try to pass from one section to the other ?

8.7.4 Now turn the **Expand** function off. Move the cursor up and down the screen. What is different ?

Reports Express

The **Expand** key can be very confusing when you first begin building reports in **Reports Express**. The [F7] and [F8] keys also need special attention. These two keys allow you to move from one section to the other. They perform different functions depending on whether **Expand** is on or off. Pay careful attention to the bottom of the screen; check to see if **Expand** is on or off before you use these keys. You will have more practice with them when you build your first report.

8.8 DATA DISPLAY FIELDS

You recognize the S and E in each one of the boxes from your experience in building forms. In **Reports Express**, the S and E mark the spaces where data will be displayed. Just as in **Forms Express**, the [F6] key locates the field. The process of naming columns or variables is the same. The report you build will contain a number of variables. Creating variables will be discussed below.

Reports Express

8.9 LEAVING THE REPORT

Leave the report by pushing the escape key twice. While you were moving around in the report you might have changed something by mistake. **Choose discard changes.**

8.10 PLANNING A REPORT

The report you viewed is a very simple one. The report you build will also be very, very simple. **Reports Express** offers you many options that we will not discuss here. Again, the **QUE R:Base** manual you have been given explains many of the more complex functions very well. We will concern ourselves with the basics.

You should begin the process of building a report just as you began building tables and forms - planning the report on paper.

Reports Express

STEP 1. Your report is based on the Table **Prinstud**.

Your first task is to decide what information you want to include in the Report. Do you want a report on grades 1 through 3? Do you want a report on all six grades? Do you want a report on boys and girls or just girls?

STEP 2. What will the report look like? Will you type out Boy, Girl, New Enrollments or will you use abbreviations? What information will appear at the top of the report? Draw the report on a piece of paper so you can get a good look at it before you begin working on the computer. To make things simple, use the same report column format as in **Classes**.

STEP 3. In order to locate data fields, you must remember the names of the table columns. Write in S and E on your report design plan [your paper] and then write the column name that will be entered in that field. Do not write anything for the fields where Totals are found. The spaces reserved for Totals will contain report variables.

Reports Express

8.10.1 REPORT VARIABLES

In the very beginning of this manual you read about the difference between input and output. You divided data collection tables in two, data input and data output. We said that output data such as totals should not be included on data collection forms because the computer would be able to do the arithmetic for you. In this section we will discuss how you design a report to create totals.

In all the other data fields in your report, you wrote the exact table column names. How do you identify the Total data fields ? As you know, there are no columns in your table, **Primstud**, that contain totals. In order to display totals in your report, you must create what is known as a **Variable**. A variable allows you to include information in your report that is not taken directly from a table. When you created a **Form**, **Express** asked you what table it would be used with. The table you named, **Primstud**, is known as the driving table. Your report will have the same driving table as your form.

Reports Express

Variables allow you to include system information such as dates, time and page numbers. Most importantly, variables allow you to compute data. In your case, you want to calculate totals. Look again at your report design. At the bottom of each column you have a space reserved for **TOTALS**. To define each **TOTALS** data field, you must tell the computer to add up all the data in each table column. Follow these steps of pre-planning to define your **Totals**.

STEP 1. Each data field holding a total represents the sum of all the data in a table column. For example, in the Classes report, the totals at the bottom of New Entrants, Boys holds the sum of all the entries in the Bnewents column in Table **Example2**. You must first give each data field a name. A sample name would be **TotNBoys**. It means total new boys. As with all other names, it cannot contain more than 8 characters. You will write the names of the variable data fields on your paper in Step 4.

Reports Express

STEP 2. Because **TotNboys** is a variable and not a column name, you must define it for **R:Base**. **TotNboys** is the sum of the column **Bnewents**. The variable is therefore defined as the **Sum of Bnewents**. **SUM OF** is a variable operator. Below you will find a list of the operators used in **Reports Express**. Ask your trainer about operators.

Figure 8.2

<i>Operator</i>	<i>Description</i>
+	Adds the value of the second operand to the value of the first operand, or combines two text values
SUM OF	Computes the sum of all values in a table column and leaves the result in the variable
-	Subtracts the value of the second operand from the value of the first operand
/	Divides the value of the first operand by the value of the second operand
*	Multiplies the value of the first operand by the value of the second operand
%	Should be pronounced "percent of." Computes the value of the first operand multiplied by the value of the second operand, then the product divided by 100.
&	Combines two text operands to form a single text string, with a single blank space between the two component parts

Reports Express

STEP 3. The **SUM OF** operator differs from the others in that it deals with only one operand, a column. The variables you will create in your first report will be totals. You will therefore use the **SUM OF** operator to define each of your variables.

STEP 4. Give each one of your **Variables** a name. Then write down the definition of each of those variables. Each name must be different. You cannot name all your variables **Total**. Follow the example below.

Variable Name-----Variable Definition

TotNboys-----=Sum of bnewents

TotNgirl-----=Sum of gnewents

etc.-----

Note: A report in R:Base can contain only Forty
Variables

Reports Express

STEP 5. Once you have completed all the planning for your report you are ready to begin building the report on R:Base. Before you begin, answer these questions:

1. Have you decided what information you want to appear in the report ?
 2. Have you drawn the report on a piece of paper to get an idea of what it will look like ? Have you left enough room for data fields ?
 3. Have you written down all the column names in the corresponding data fields ?
 4. Have named all your variables and defined them ?
-

STEP 6. If you answered yes to all of the above, you are ready to begin building your report.

Reports Express

8.11 CREATING A REPORT

With your report plan before you, begin your report by choosing **REPORTS EXPRESS** from the **R:Base System V** main menu.

STEP 1. The **Reports Express Main Menu** should be on your screen.

```
Reports EXPRESS Main Menu
(1) Choose a database
(2) List directory
(3) Change directories
(4) Exit from Reports EXPRESS
```

CHOOSE OPTION 1, CHOOSE A DATABASE

STEP 2. You are now looking at the **Choose a Database Menu.**

```
Choose a database
TCAIRO STAFF EGYPT
```

CHOOSE TCAIRO

Reports Express

STEP 3. The Report Options Menu is now on your screen.

Report Options Menu
(1) Edit/Create a report
(2) List reports and associated tables
(3) Copy a report
(4) Remove a report
(5) Return to Reports EXPRESS Main Menu

CHOOSE OPTION 1, EDIT/CREATE A REPORT

STEP 4. The Choose a Report Menu is now on your screen. You are creating a new report.

Choose a report
Classes (New)

CHOOSE (NEW)

STEP 5. You are now asked to name your report.
Enter the name PrimRoll. It stands for
Primary Enrollment.

Enter your report name (1-8 characters)

Reports Express

STEP 6. Once you have typed in the name of your report the **Select a Table or View** menu will appear. You are asked to choose the table which drives your report.

Select a table or view		
Example1	Example2	Primstuf

CHOOSE PRIMSTUF

STEP 7. You now have the **Reports Definition Menu** on your screen. There are four menu choices. They are explained below.

EDIT- will allow you to enter the work area of the screen where you can begin building your report. If you were working on an existing report, you would also choose **EDIT** to enter the report. You used the **EDIT** option to enter the **Classes** report in the previous section.

Reports Express

EXPRESSION- can be used to define your report variables. You are not required to use this option to define your variables, however. You will be given the chance to define variables once you begin building your report.

CONFIGURE- option is used to set up page formats and to set breakpoints. **Breakpoints** are used when your report contains subtotals. We will not use this menu option now. Ask your trainer about this option when you build more complicated reports.

DRAW- The draw option is used to draw and erase single or double lines on your report. This menu choice is completely optional. Your reports and forms look much nicer if you use the **Draw** option. Once you are more comfortable with the other more complicated and important functions go back and experiment with this option.

Reports Express

STEP 6. Choose each one of the menu options in order to see what their menus look like. First choose the **Expression** option. You will see the **Expression Menu** on your screen.

Expression Menu
Define Delete Retype Reorder

You have not created any variables so the choices this menu offers will not help you. Once you have finished your report you can go back and look at these options. There is nothing to see right now. Press **Escape** to leave the menu.

Reports Express

STEP 9. Choose the **Configure** option. The following screen will appear. Once again, you can not use any of the options until you have created your report. Press **Escape** to leave the screen.

```

Lines Per Page .....: 60
Remove Initial Carriage Return..: [NO ]
Manual Break Reset .....: [NO ]
Page Footer Line Number.....: 0
    
```

	BREAKPOINTS		FORM FEEDS			
	Break Column	Variable Reset	Header		Footer	
			Before	After	Before	After
Report Page		[NO]	[NO]	[NO]	[NO]	[NO]
Break1	[None]	[NO]	[NO]			
Break2	[None]	[NO]	[NO]			
Break3	[None]	[NO]	[NO]			
Break4	[None]	[NO]	[NO]			
Break5	[None]	[NO]	[NO]			
Break6	[None]	[NO]	[NO]			
Break7	[None]	[NO]	[NO]			
Break8	[None]	[NO]	[NO]			
Break9	[None]	[NO]	[NO]			
Break10	[None]	[NO]	[NO]			

Reports Express

STEP 10. Choose the **Draw** option. The **Custom Border Menu** will appear. Press **Escape** to leave.

```
Custom Border Menu
Sing) Lines Double Lines Erase
```

RH

STEP 11. Choose **EDIT**. You now have a blank screen with a white stripe on the left hand side. To become familiar with creating sections, first make sure that the **Expand** option is on. [Look at the bottom of your screen.] If it is off, switch it on using the **[F9]** key.

STEP 12. At the top of the white strip you see an **RH**. As you know this means **Report Header**. Type in any titles and descriptive information you want to display at the beginning of the report. Copy it from your report design on paper. Make sure the **Expand** option is on or you cannot create more than one line in **RH**. With **Expand** on you can create new **RH** lines by pressing the return key or using the cursor to move down.

Reports Express

STEP 13. When you have finished writing the report header, press the **[F8]** key. This will create the next section, **PH**. **PH** stands for **Page Header**. This is the section where you should create the column titles, e.g. New Entrants, First Grade etc. Type in the design of your report as you have drawn it on paper. Remember that you can make your report 255 columns wide. If you run out of room on the right of the screen there is no problem. A new screen will appear and you can continue your work. Simply use the cursor keys to move left and right in your report.

STEP 14. Do not worry about drawing in lines with the **Draw** option right now. You can go back and edit your report later. When you have finished creating the column titles push the **[F8]** key and you will create the next section, **D**. **D** stands for detail. You will create the data fields in this section. Remember you only need to create one data field for each column in the table. **R:Base** will automatically create a row in the report for every row it retrieves from the driving table.

Reports Express

STEP 15. Remember the rhythm of creating data entry fields in **Forms Express**. Refer to your design on paper and begin locating fields. If you want to double check the column names push **[F3]** and all the driving table's columns will be listed for you.

Move the cursor to where you want to locate the first field. Press **[F6]**. The following message will appear.

```
*****  
Column or variable name:  
RH  
RH  
RH  
PH  
PH  
PH  
PH  
*****
```

Reports Express

STEP 16. Type in the name of your first field. You should not have to define an expression because it is a column name. If **R:Base** asks you to define an **Expression** then you know you have not typed in the exact name of a column. If that happens, push **[F3]** to double check the column names. If you typed in the correct column name then **R:Base** will ask you to locate the field with the following message.

```
*****  
Move cursor to start location and press [S]           Locate:TotNboys  
RH  
RH  
RH  
PH  
*****
```

Move to the exact location you want to begin your data field and press S. Just as in **Forms Express**, S means Start. Now press E, for End, to end the data field.

Reports Express

STEP 17. When you have finished locating all your data fields you must create the next screen section, RF. Press [F8] and PF will appear. PF stands for **Page Footer**. Your report is a simple, one page report. You do not need to write anything at the bottom of your report page. You do, however, want to create **Totals**. Your totals should go in the RF section of the screen. RF stands for **Report Footer**. This means that whatever is in the RF section of the report is the last thing that appears in the report. This is where you want your totals to appear. Push [F8] again. PF will change to RF.

STEP 18. Move the cursor to where you want to locate your first variable. Press [F6]. At the top of the screen you will see the following message.

Column or variable name:

RH
RH
RH

Reports Express

STEP 19. Reports Express is asking you to define an expression. An expression is a variable definition. You have already defined them on paper in your report design plan.

Earlier we said that all your variables will be **Sums**. Look at the report design you have written on paper and type in the definition of your first variable. For example, if the name of your first variable is **TotNboys** your variable definition would look like Figure 8.3.

Figure 8.3

```
Define Expression:TotNboys =  
RH  
RH  
RH
```

Reports Express

STEP 20 Once you have typed in the expression, press the enter key and **R:Base** asks you to **Move the cursor to start location and press [S]**.

```
*****
Move cursor to start location and press [S]                               Locate:TotNBoys
RH
RH
RH
PH
PH
PH
PH
D
D
*****
```

Move the cursor to where you want the data field to begin and press S. Press E where you want it to end.

STEP 21. Repeat the process until you have located all your report variables. When you have finished, push **Escape**. The **Reports Definition Menu** will appear. Press **Escape** again. This menu will appear.

```
*****
Report Exit Options Menu
Save Changes Discard Changes Return
RH Report on Classes and Students
RH Database TCairo
*****
```

Reports Express

Choose **Save Changes**.

STEP 22. You are probably very tired. Take a break. Work your way out of **Reports Express** by pushing the **Escape** key. When pushing **Escape** does not give you another menu, but only rings a bell, choose option 4, **Leave Reports Express**. When you see the **R:Base System V Menu**, choose the last option, **Return to Dos**. You are now out of R:Base.

SECTION 9

R:BASE COMMAND MODE

9.1 SECTION OBJECTIVE-----

1. The Participants will have a basic understanding of R:Base Command Mode demonstrated by their ability to execute a series of operations using the OPEN, ENTER, LIST, HELP, OUTPUT, SORTED BY, WHERE and SET VARIABLE commands.

9.2 INTRODUCTION-----

R:Base Command Mode differs from the **Express** mode in that it does not provide you with instructions or menus. Before using command mode, you must learn a series of commands to be typed in for every operation you want to execute. The commands are fairly simple and do not take much time to learn. If you make a mistake the screen will show you the correct syntax for the command. Once you have learned the commands you will be able to perform a wide range of **R:Base** operations that are not available to you in the **Express** mode.

9.3 ENTERING COMMAND MODE

STEP 1. In order to enter the **R:Base Command Mode** you must work your way back to the **Main** menu. Do this by pushing the escape key on your key board. The escape key will take you to the previous menu. Keep pushing escape until no new menu appears and a bell rings; you will then have to choose the last option on the menu -- exit.

R:Base Command Mode

STEP 2. Once you arrive at the main menu, choose **R:Base** and then push the escape key twice. The **R>** prompt should appear in the left hand corner of your screen.

STEP 3. The first command you need is the **OPEN** command. Simply type **OPEN** and then the name of the database you want to open. In this case it will be **ICairo**.

```
R> Open _____
```

STEP 4. The **R>** prompt will read **Database Exists;** you are ready to type in a new command.

9.4 LOADING DATA

The first action we want to perform in command mode is loading data using a data entry form. The command for entering data is the **ENTER COMMAND**.

R:Base Command Mode

STEP 1. What form are going to use ? Write the name of the form on the blank line below.

R>Enter _____

STEP 2. Push return and the form you have chosen will appear on the screen. The name of the form you built is **Prinstud**. That is the form you should view on your screen.

STEP 3. When the form **Prinstud** appears on the screen, the cursor will be on the first data entry field. Enter the data from Figure 9.1 into the highlighted area and press return. The cursor automatically moves to the next field. Type in the appropriate data. It becomes clear why you should pay attention to the order you locate the fields when you are building your form.

R:Base Command Mode

Figure 9.1-----

R:Base Command Mode

STEP 4. When you have filled in the last entry the screen will display a menu with the cursor on **Add**. Push the return key and the form will appear blank once more. The data you typed in has been added to the database tables. Each time you complete a form, you add another row to the database. If there are any categories that have no data, leave the data entry field blank and **R:Base** will automatically insert a 0.

STEP 5. When you have entered all of your data, choose **QUIT** from the menu above the form. You will then be back in command mode.

9.5 PRINTING A REPORT

STEP 1. To remind yourself of report names, type in
R>List Reports.

STEP 2. Begin by looking at the on line help screen
for the print command by typing in **R>Help
Print.** The following screen will appear.

PRINT	
Use PRINT to generate a report defined in Reports EXPRESS.	
• If the report contains breakpoints, rows are automatically sorted on the breakpoint columns.	
• Use a SORTED BY clause to sort the data by up to 10 columns. The SORTED BY clause overrides breakpoints.	
• Use a WHERE clause to specify the rows that you want to print.	
SYNTAX	
PRINT rptname	[SORTED BY collist] [WHERE condlist]

R>

R:Base Command Mode

STEP 3. Look at the lower of the two boxes. This box outlines the **Print** command for you. It is known as the syntax box. Similar boxes will appear whenever you type **HELP** followed by a command.

STEP 4. Before you print your report you should tell the computer where you want the report to appear. You can send the report to the printer, the screen, both screen and printer or a disk. The command for these options are as follows.

1. **OUTPUT PRINTER-** tells the computer to send the Report to the printer. Once you enter the **Print** command the report will be printed on paper.
2. **OUTPUT SCREEN-** tells the computer to show you the report on the screen.
3. **OUTPUT filespec-** tells the computer to send the report to a disk. The report can then be printed later.

R:Base Command Mode

Type in **R> Output screen**. We will print other reports on paper later.

STEP 5. You are now asked to type in the **Print** command then specify the report name. Write in the report name you want to print on the blank line below.

R>Print _____

This command alone will print your report. You have other options however. The **Sorted By** command allows you to print a report with the columns you specify sorted in alphabetical or numerical order. The **Where** command allows you to print only certain rows of the report instead of the whole report. More detailed explanation of these optional commands can be found in the section on **Data Retrieval**.

STEP 6. Entering your print command will signal the computer to send your report to the screen for you to view.

9.6 INITIALIZING YOUR VARIABLES

The report that you created contains a number of variables. Those variables add up the totals of certain columns. Each time you print a report, you must make sure that the report variables are put back to zero. If you don't do this the totals will not be correct.

To illustrate the point, print your report again using the same command as above. Notice that the totals are incorrect.

The command used to initialize variables is **SET VARIABLE**. R:Base command mode requires that you type in the **Set Variable** command and then the report variable name. You must type the command for each variable that appears in your report. Write the name of your report variables in the blank lines below to form the commands you need to initialize your variables.

R>Set variable_____ To 0

R>Set variable_____ To 0

etc.

Note: If you forget the Variable names, type **List Variables** at R> prompt.

R:Base Command Mode

Print your report again and check to see if the totals are correct.

9.7 CONCLUSION

You are probably thinking about how much work it is to initialize all the variables every time you want to print a report. You are quite right! There is a much faster way to print standardized reports. Reports you have built can be selected from a menu you build in **Application Express**. Before we turn to building an application, let's look closer at the report you have already built.

To Exit from the system, Type **Exit**

SECTION 10

RESTRUCTURING THE DATABASE

10.1 SECTION OBJECTIVES

1. The participants will gain a deeper understanding of how **R:Base** relates the data stored in different database tables by evaluating the report they have built for usefulness in planning.
2. The participants will increase their knowledge of database construction demonstrated by their ability to design sample data collection tables which reflect clear data collection objectives. The design of the tables must also allow for the execution of logical operations within the **R:Base** package.

Restructuring The Database

3. The participants will develop a basic proficiency in some of the **RiBase** commands which allow for changes in the database structure demonstrated by their ability to successfully use the **Expand, Build Key, Delete Key, Redefine and Remove Commands.**

10.2 DATA COLLECTION

In professor Davis's lecture on planning models, section 1.6.1, you will find mention of " the structure or parts of an institution or school system; how the faculties and departments and the central administration relate to each other; or how the national ministry relates to the education offices of the governorates or provinces; and how these in turn relate to the school districts and individual schools." Data collection tables are used to collect data from many different levels within the education structure.

Restructuring the Database

A systems chart shows the various levels and types of schools in a national system. For example, data collection forms may be designed to collect enrollment data for those various levels. Ministries are also interested in age, grade and sex data. The types of data that can be collected are numerous.

All data is not useful in all sectors of educational planning, however. One must consider the reasons for collecting data and, once collected, the part it plays in educational planning. The purpose behind collecting data must be clear before you begin designing a database around it.

Restructuring The Database

10.3 EVALUATING TABLES, FORMS AND REPORTS

Call up your report on the screen. As you examine it, answer the following questions.

1. What information does the report provide you with?
 2. How many schools are included in the report?
 3. What school does the report describe; what zone; governorate ?
-

Unless you have a very active imagination, you can't truthfully answer these questions. It is important to remember that the example table was taken from a set of tables reporting on the primary school level. Your next task is to analyze the different tables in the form and restructure your table so that the data it contains can be linked to other data tables.

Restructuring The Database

STEP 1. In Section 6, **Pre-Design and Creating Tables**, we discussed possible changes in the data collection form. Take out your notes from that section and look them over again.

STEP 2. What are the changes you suggested for the tables ? Why did you suggest them ?

10.4 STANDARDIZATION OF DATA

The front cover of the data collection form you have been given shows vital school information such as its name and location. Because all tables appear on the same folded piece of paper, you know that the data they contain all refers to the same school. How will **R:Base** know that ? Unless you include a column in each database table that somehow connects the data to a governorate, zone or school, neither **R:Base** nor you will ever know where the data comes from.

To get an idea of how you can link tables within a database together, we will look at an updated version of table **Example2**.

Restructuring The Database

STEP 1. Go to the **R:Base System V Menu** and enter **Definition Express**. Choose [2] **Modify a Database**.

STEP 2. Open the **Egypt Database** and then choose **Tables**. Choose option [2].

STEP 3. Choose **Update2**.

STEP 4. Move through the table and write down the differences you notice between **Example2** and **Update2**.

Update2 contains columns that supply the database user with important descriptive data. . By assigning a standardized code to each governorate and school, a link between all tables has been established. All tables have at least one column in common. This allows **R:Base** to perform a wide range of relational and logical tasks. Data bearing the same identification codes can be identified and added, deleted or changed within tables as well as between different tables. Data can be compared, summed and computed. All these operations depend upon a reliable, standardized link between tables.

Restructuring the Database

10.5 THE NEW COLUMNS

There are four new columns in table **Update2**. They are:

Govcode- which stands for Governorate Code. A database user may want a report which arranges the data by governorate. Having this code allows him to do so. An example would be a Ministry official who wants a list of the schools in Governorate X.

City- column would give the city in which the school is located. Reports could then be drawn up on schools arranged by city.

Schlname- stands for School Name.

Schoolid- stands for School identification number. It will allow reports to be generated on an individual school level.

Restructuring the Database

10.6 ADDING NEW COLUMNS USING EXPAND

You are able to add new columns to your table using the **Expand** command. This command must be typed while in **R:Base** command mode.

STEP 1. Enter the **R:Base** command mode and open the **TCairo** Database.

STEP 2. The syntax for the **Expand** command follows:

SYNTAX
EXPAND tblname WITH colname [=expression] [datatype [length]]

R>

In order to expand your table, **Primstud**, you must type in the table name, the column names and the Datatype. Below you will find the **Expand** command syntax with blank lines where you must fill in the needed information. You are going to expand the table with four columns so you will have to repeat the command four times.

Restructuring The Database

Table Name	Column Name	Data Type
R>Expand_____	With _____	_____

The data type specification refers to the same choices of data types you had when you created Primstud in **Definition Express**. The two columns containing id. numbers are integers and City and Schlname contain textual data. Write in Integer on the blank line after the two columns that contain numbers. For the other two columns, write in Text. When you create columns that hold textual data, you must also tell the computer how much space you want to assign to each column. If the name of a school usually has 10 letters, then you would type in 10 after the Text entry. If you think it needs more, type in a higher number of spaces.

Restructuring The Database

10.7 MORE ON STANDARDIZATION

In the previous exercise, you added columns to the tables using the **Expand** command. Columns can also be removed from tables using the **Remove** Command. The syntax is similar to the **Expand** command. The remove command can also be used to remove forms, reports, rules, tables or views. The syntax can be viewed by entering R:Base command mode and typing in **Help Remove**. The following screen should appear.

REMOVE	
Use REMOVE to delete a column, form, report, rule, table, or view from the database.	
SYNTAX	
REMOVE	COLUMN colname FROM tblname
	FORM formname
	REPORT rptname
	RULE rulenum
	TABLE tblname
	VIEW viewname

R>

We talked about standardizing data in section 10.4. The columns you added which contain identification codes are examples of including standardized data in your database. Look at the two code columns again.

Restructuring the Database

There are two different codes. If you devise a standardized code that identifies both the governorate and the school you would only have to work with one identification number instead of two. For example, if you have 10 governorates in your system you could assign a prefix code for the governorate and then assign a school code. The codes might look like this.

<u>Governorate Name</u>	<u>Governorate Code</u>
Fes	10
Rabat	20
Beni Mellal	30

The second part of the code would identify the school. All schools whose number began with 10 would be in the Fes governorate. All schools whose number began with 20 would be in the Rabat governorate. All schools whose number began with 30 would be in the Beni Mellal governorate.

Restructuring the Database

The identification codes might look like this.

<u>School Name</u>	<u>School Code</u>
Moulay Idriss	10100
Moulay Rachid	10101
Ibn Khaldoun	20100
Sidi Youssef	20101
Ben Khayat	30100
Medina School	30101

When you see the code 30101, you would know that the school is in the Beni Mellal governorate. You could then call for a list of all the schools in that governorate. You may want to design a code that identifies the city as well but you could always leave the city name in the table as well.

Eliminate the unnecessary governorate column using the **Remove** command.

Restructuring the Database

Your database is still very small but once it becomes a huge list you will want to make certain column keys. The school identification code would make an excellent key because it will contain unique data. To make a column a key use the **Build Key** command.

Syntax For Build Key

R>Build Key For "column name" in "Table Name"

Keys can be deleted just as easily if you decide that you want to change a key column. Use the **Delete Key** command to remove a key from a column. The syntax is the same as in the **Build Key** command.

R>Delete Key for "column name" in " Table name"

Restructuring the Database

10.10 UPDATING FORMS AND REPORTS

Now that you have made changes in your table, you will have to update your data entry form and your report. To get an idea of how your new form and report might look, view the form **Update2** and the report **PPenroll** in the **Egypt** Database. To make sure you do not totally erase your original work by mistake, choose the Copy option from the menus in **Form and Reports Express** and make copies of your work and then edit your copies. You will have to give new names to the forms and reports to make copies of them. The copy menu option gives you the syntax and all you need to do is type in the present name of the form and report and then a new name.

10.11 ADDING DATA TO THE DATABASE

Add data to the revised database using the **Enter Command** in **R:Base Command Mode**. Use the data in Figure 10.1.

SECTION 11

APPLICATION EXPRESS

SECTION OBJECTIVES

1. The participants will be familiar with the basic concept of what **Application Express** does by running a sample application.
 2. The participants will become familiar with the **Application Express** menus by viewing menu options.
 3. The participants will understand the steps involved in building an application demonstrated by their ability to build a sample application using the table, form and report they have built.
-

Application Express

11.1 INTRODUCTION

You have now seen a series of commands to create tables, forms and reports. You have altered the basic structure using **R:Base** commands and have printed out a report. Earlier we mentioned how much work it was to initialize all the report variables. You have learned quite a few commands by now and it is impossible to remember all of them off the top of your head. The **R:base** QUE manual you have been given will be a great help when you need to look commands up.

There is a much easier way to move around in the database once you have built the basic structure. You can avoid long series of commands to load forms, initialize variables and print reports by building an application in **Application Express**. In the very simplest of terms, application express allows you to design your own menus. You can then use these menus to add data, change data, delete data, ask questions about the data and print reports. Using an application to print reports saves you all the time you spent initializing variables the first time you printed a report. Your report only had a small number of variables. Imagine if you had to type `Set Variable to 0` forty times every time you wanted a report!

Application Express

11.3 LOOKING AT AN APPLICATION

STEP 1. Enter **R:Base command mode** and open the **Egypt** database.

STEP 2. The command for running an application is
R>run name_of_application in name_of__application.apx. The application that has been built is called **Pre-prim**, for **Pre-Primary**. Type in the following command:

R>run pre-prim in pre-prim.apx

STEP 3. The following menu will appear.

```
Pre-Primary School Data Main Menu
(1) Maintain School Data
(2) Print Pre-Primary Reports
```

Application Express

STEP 4. The **Pre-Primary** menu gives you two options:

1. **Maintain School Data-** allows you to add new data, delete data or manipulate the data in a variety of ways.

2. **Print Pre-Primary Reports-** allows you to print reports without any of the work involved in initializing variables.

STEP 5. Choose **Maintain School Data** and the following Menu will appear.

<p>Maintain Pre-Primary School Data</p> <ul style="list-style-type: none">(1) Add A New Pre-Primary School To The Database(2) Change Data Of A Pre-Primary School You Have Already Entered(3) Remove A Pre-Primary School From The Database(4) Ask Questions About a Pre-Primary School In The Database
--

You have four option from which to choose:

1. **Add a New Pre-Primary School to the Database.** Choosing this menu will produce your data entry form. You can then add data to the database.

Application Express

2. **Change Data of a Pre-Primary School you have already entered.** Choosing this option will allow you to change data using the data entry form.
3. **Remove a Pre-Primary School from the Database.** Choosing this option will allow you to delete data from the database.
4. **Ask questions about a Pre-Primary School.**
This option allows you to query the database for specific information.

All these menu choices were created in **Application Express**. You can design menu titles and messages that the user will then be able to choose from. It becomes obvious that pre-planning is vital to building an application. You must have a clear idea of what you want the application menus to offer.

Return to the R:Base System V Main Menu and exit the system or continue on to next section.

Application Express

11.4 BUILDING AN APPLICATION

In this section you will build a sample application similar to the one you just ran. Building an application for one table, form and report is not all that realistic but it will allow you to practice using application express on a simplified level.

STEP 1 Choose **Application Express** from the Main Menu. The following menu will appear:

Application EXPRESS Main Menu (1) Define a new application (2) Modify an existing application (3) DOS functions (4) Exit from Application EXPRESS

STEP 2 You are faced with four choices.

1. **Define an new application-** allows you to build an application.

Application Express

2. **Modify an Existing application-** allows you to make changes in an application you have already built.
3. **Dos Functions-** allows you to access a limited number of **Dos** commands and directories.
4. **Return to the Express Menu-** returns you to that menu.

Choose option 1.

STEP 3 Choose the database where your work is found.

STEP 4 Name your application **Primary**

STEP 5 Accept the name **Main** by pressing return.

STEP 6 Choose **Vertical**. This choice will allow you to write longer menu choices as you saw in **Pre-Prim**. The **Horizontal** menu allows only short option descriptions.

Application Express

- STEP 7** Look at the menus in Steps 3 and 5. In the next step you will be asked to supply titles and menu options. Type in the same titles and options in Steps 3 and 5 but change Pre-Primary to Primary.
- STEP 8** Now you must give the menu a title. **Pre-Prim** used **Pre-Primary School Data Main Menu**. Type in a similar menu title and enter it in.
- STEP 9** Type in the Menu options you want the user to be able to choose from. **Pre-Prim** had two options. Type in your menu choices. Press **Escape** when you have finished.

Application Express

STEP 10 Choose **Yes** when asked if you want to use the escape key to exit from menus. Using escape will put you in the previous menu.

STEP 11 You are now asked if you want to create a help screen. Answer **No** for now. You can always build one later using the modify an application option.

STEP 12 You must now give more specific information on each one of your menu choices. **Express** asks for information for each option one by one. The first option is **Maintain School Data**. You need to establish a link to a submenu. Choose **Menu** from the horizontal menu at the bottom of the screen.

Assign actions to menu option 1

Choose an action
Load Edit Delete Modify Select Print Custom Macro Template
Menu Password Exit

Application Express

- STEP 13** Express asks for a name. Type in **Prim**.
Choose the **Vertical** menu. Type in a menu title. Type in **Maintain Primary School Data**.
- STEP 14** Enter in each menu option. **Add, Change, Remove and Inquire**. You can add directions to the rest of the menu option as you saw in the **Pre-Prim** application. For example, the menu option for add did not just say add. It read **Add a new Pre-Primary school to the Database**.
- STEP 15** When you have typed in the four choices, press escape and **Express** returns to the **Main Menu**. **Express** asks if you want another action for this menu. Answer **No**.
- STEP 16** Express moves to the **Print** option on the main menu. Choose **Print** when asked to assign actions to menu option 2.
- STEP 17** Select **Primstud** from the **Tables** menu.
Select your report from the **Reports** menu.

Application Express

- STEP 18** Respond **No** when asked if you want to edit the report. Select **Schoolid** as the sort column. Choose Ascending sort order. Push **Escape**. When asked to choose columns to validate, choose **Schoolid** and **Exists** as the operator. This will print all the schools that are in your database. You may not want to do this in all reports but you can make changes later. Press **Escape** when finished. Answer **No** when asked if you want another action. Answer **YES** to the escape key question. Answer **NO** to Help screen.
- STEP 19** The **Maintain Primary School Data Menu** appears. The cursor is on the first option, **add**. When you are asked to select an action from the menu, **choose Load**. This will signal the computer that you want to load data into a specified table.
- STEP 20** **Express** now asks you which table the load command refers to. Respond by choosing **Primstud** and pressing enter.

Application Express

- STEP21** You will now see all the data entry forms you have created. Choose **Primstud**. Answer **No** when you are asked if you want to edit the form. When you are asked if you want to specify another action, answer **No**.
- STEP 22** Express will move to the next menu option, **Change....etc**. Choose **Edit** from the choice of 12 menu options listed.
- STEP 23** Again you are asked what table the command refers to. Choose **Primstud**. The next screen shows the forms menu. Choose **Primstud**.
- STEP 24** Answer **No** when asked if you want to edit the form.

Application Express

STEP 25 You must now specify the columns you want the user to edit. You cannot know all the editing needs of the person using the database so you should choose one column which will allow the user access to all the columns. The best column to choose is the Schoolid column. Press **Return**.

Now choose **Ascending Order**. Push **Escape** at the next screen.

STEP 26 A menu containing comparison operators will appear at the bottom of the screen. A comparison operator is used to compare the data in a column to another value. In this case you want the user to be able to identify a specific school by its identification number. Choose Schoolid. Then choose **EQ**. **EQ** means equal to. When the user types in a school id number he is asking to see all the data connected with that school, or equal to the id number.

Application Express

STEP 27 Express asks if you want to enter a comparison value. Answer **Yes**. Now it asks for a prompt message. Answer **Yes**. Think of a message. For example, Enter Id Number of School Editing. Any message that makes sense and fits is acceptable. Remember that you want the user to enter the school identification number; any message that prompts him to do that is good. Push **Escape**.

STEP 28 Do not enter any more logical operators. Choose **Done** from the menu. Answer **No** when asked if you want another action.

STEP 29 Express moves on to **Remove a School**. Select **Delete** from the list of 12 Actions. Choose the Schoolid column and the EQ operator just as you did for the edit command in the previous menu when asked what column and operator you want to use to delete data. Follow the same steps to complete this operation as you did above.

Application Express

STEP 30 The next menu option, Inquire about a school's data, allows the user to look at data without make any changes. The series of menu choices follows the same rhythm as the other menu options. Choose the **Select** command from the action menu. It is impossible for you to know how many different questions the database user might want to ask. It, therefore, makes sense to display general data that the user can use to form more specific queries using the **Select** command in the **R:Base** command mode.

STEP 31 Following the same rhythm, choose the **Primstud** table when asked. You must then choose the columns you want to display. Choose **Schlname**, **Schoolid** and **City**. This information will be useful to someone who wants to find out the names and id numbers of schools in a particular governorate. As in the other operations, push escape when done.

Application Express

STEP 32 When asked for a **Sort** order, Choose **Schoolid**. Choose **Ascending**. Push **Escape**. To validate, choose the operator **LE**. **LE** means less than or equal to. If the user did not find the school he wanted in the first data supplied he could enter the next governorate id prefix plus three digits and all the data in the first two governorates would be shown. For example, if the user typed in 20000 at the prompt, all the schools in the first governorate would be shown because they all have an id number below 20000.

Follow same steps as above concerning prompt messages. Press **Escape** when finished.

Answer **No** to all other questions. Leave the system by working your way back to the **Main R:Base System V Menu**. Choose **Return to Dos**.

SECTION 12

MORE R:BASE COMMANDS

AD HOC QUERIES

LOGICAL OPERATIONS

12.1 SECTION OBJECTIVES

1. The participants will increase their knowledge of R:Base commands by successfully executing a series of ad hoc queries.
 2. The participants will demonstrate an overall knowledge of the R:Base System V package by beginning work on a database design pertinent to their own professional needs. Such designs may be as basic as a series of questions or as sophisticated as actual database development.
-

12.2 INTRODUCTION

You have already worked with a number of **R:Base** commands. There are many more to learn. To illustrate a number of **R:Base** ad hoc query commands, we will look at a new database which contains employee records. Ad hoc queries are individual questions that you ask the computer. In the section on **Reports Express** you created reports which presented a range of informational statistics. Ad hoc queries are typed in to the computer one by one. They are very specific questions.

We will also explore exercises in some of the logical, or relational, operations in **R:Base System V**. Relational operations allow you to add tables together, edit two tables and join them or copy tables and then edit them. These are just a few of the many operations you can perform in **R:Base command mode**. The key to these operations is planning. The actual commands are very simple. The concept behind why you perform them is more complicated.

Command Mode Exercises

12.3 LOOKING AT A PERSONNEL DATABASE

Up until now we have been looking at tables and forms which contain enrollment data. **R:Base** is also capable of storing data in textual form. This means, for example, that you could build a database of all the teachers who work for a Ministry of Education, what they teach, how much they are paid, et cetera. Such a database could be very useful to planners for a number of reasons. It could be used for stocktaking purposes or financial planning. We will now look at an example of a personnel data base.

STEP 1. Begin by choosing **Definition Express** from the **R:Base System V Main Menu**.

STEP 2. Instead of choosing the **TCairo** database, choose **STAFF**.

STEP 3. Choose the table named **Tpersinf**. **Tpersinf** stands for **Teacher/Personnel Information**. The columns in the table hold data consisting of names, salary, contract numbers et cetera. Move through the table to view all the columns.

Command Mode Exercises

STEP 4. To understand what all the column names mean, exit from **Definition Express** tables and go to the **Forms** menu.

STEP 5. Choose the form named **Tstafinf**. **Tstafinf** stands for **T**eaching **S**taff **I**nformation.

Command Mode Exercises

STEP 6. Choose the **Edit** option. You can find out which column name corresponds to each data entry field by moving the cursor in between any S and E on the form. Each S and E is next to a phrase. Once the cursor is in between the S and the E, push the F3 key and the screen will display the column name corresponding to that data field. In this way, you will see the connection between the abbreviated column names and the data entry form design.

NOTE: Some of the entries will be codes instead of long textual entries. This enables the computer to work faster because the data is standardized. Retrieving data becomes easier for you because you will be using the same standardized codes in many tables. A list of the codes and their meanings is on the following page.

Codes For Personnel Database

Status Codes

Executive Appointment = 01
 Supervisory = 02
 Teaching = 03
 Full Time = 04
 Part Time = 05

Positions

Director
 Headmaster
 Deputy Headmaster
 Senior Teacher
 Teacher
 Senior Librarian
 Librarian
 Senior Counsellor
 Counsellor
 Administration/Clerical
 Senior Lab Technician
 Technician
 Technical
 Service
 Other

_Subject Codes [Preparatory]

Classroom Teacher .01
 Arabic Language/Religion 01
 English Language 02
 French Language 03
 Mathematics 04
 Science and Health 05
 Social Sciences 06
 Art 07
 Physical Education 08
 Agriculture 09
 Music 10
 Home Economics 11
 Practical Studies 12
 German Language 13
 Supervisory Only 14

Levels

Pre-Primary = Pre-Prim
 Primary = Primary
 Prepare = Preparatory

Grades

Pre-Primary First Grade = .001
 Pre-Primary Second Grade = .002
 Primary First Grade = .01
 Primary Second Grade = .02
 Primary Third Grade = .03
 Primary Fourth Grade = .04
 Primary Fifth Grade = .05
 Primary Sixth Grade = .06
 Preparatory Seventh Grade = .07
 Private Preparatory Seventh Grade = .007
 Preparatory Eighth Grade = .08
 Private Preparatory Eighth Grade = .008
 Preparatory Ninth Grade = .09
 Private Preparatory Ninth Grade = .009

Command Mode Exercises

12.4 LOADING DATA

STEP 1. To load data using Tstafinf return to R:Base
Command mode.

STEP 2. Open the Staff Database with the open
command.

R> Open _____

STEP 3. Use the Enter command to display Tstafinf.

R> Enter Istafinf

STEP 4. The following data will form the
basis of your Personnel data base. When
loading the data make sure to pay
attention to the order of data entry; only
the first example includes data headings.
[Data to be entered is underlined]

Command Mode Exercises

Data to be loaded into table Ipersinf using the form:
Istafinf:

School Year: 8687

Last Name: Masrou First name: Mustapha

Sex: Male

School Identification Number Where Employee is Registered:

00137 *

School Identification Number Where Employee is Working:

00137 *

Contract Number of Employee: 751

Salary: 1000

Position: Director

Status: 01

School Level: Prepare

Grade:

Subject Taught: 14

* Not all employees work in the school where they were originally assigned. By documenting the two identification numbers, R:Base will be able to count the number of teachers who are not working in their originally assigned schools.

Command Mode Exercises

Using the Form `Tstafinf`, load the following data into the database.

8687		8687	
Benjelloun, Najib		Mansour, Hakima	
Male		Female	
00137		00137	
00215		00137	
753	200	752	200
Teacher		Teacher	
04		04	
Prepare	07	Prepare	09
03		02	

Command Mode Exercises

8687		8687	
Kajani, Khadija		BenHadid, Abdelkarim	
Female		Male	
00215		00215	
00215		00317	
754	400	755	600
Senior Teacher		Deputy Headmaster	
02		03	
Prepare	08	Prepare	07
04		01	

8687		8687	
Alaoui, Nadia		Lahrichi, Mohammed	
Female		Male	
00215		00137	
00215		00137	
756	800	757	200
Deputy Headmaster		Teacher	
02		04	
Prepare		Prepare	07
14		08	

Command Mode exercises

8687		8687	
Lahrichi, Majid		Benjadid, AdelNasser	
Male		Male	
00137		00137	
00137		00137	
758	200	759	200
Teacher		Teacher	
04		04	
Prepare	09	Prepare	07
08		06	

Command Mode Exercises

8687		8687	
Keliani, Samir		Bensallik, Miriem	
Male		Female	
00137		00137	
00137		00215	
760	900	761	300
Headmaster		Senior Librarian	
01		04	
Prepare	04	Prepare	05
14		14	

8687		8687	
Kettani, Anissa		Nadour, Youssef	
Female		Male	
00137		00215	
00215		00137	
762	200	763	300
Librarian		Senior Counsellor	
03		03	
Prepare	.007	Prepare	.008
05		12	

Command Mode Exercises

8687		8687	
Aboutaib, Jalil		Sefiani, Amina	
Male		Female	
00137		00215	
00137		00215	
764	100	765	300
Counsellor		Senior Lab Technician	
05		03	
Prepare	.008	Prepare	.009
07		05	

8687
Toufik, Amir
Male
00215
00215
765 200
Technician
03
Prepare 07
03

Command Mode Exercises

12.5 Data Retrieval/ Asking Questions

Now that you have loaded some data into the database you will be able to retrieve that data in a variety of ways. In order to retrieve data without using the Reports function you must again enter the **R:Base** command mode.

STEP 1. Open the **Staff** database using the **Open** command.

When you want to retrieve and present standardized data, it makes sense to use the Reports Function of **RBASE System V**. There are times, however, when you want to retrieve limited and specific data. The fastest and most efficient way to retrieve such data is through use of the command mode. We will first look at data stored in Table **Tstafinf**. Before you begin the exercises, you should familiarize yourself with the syntax commands for data retrieval. If you make an error, the screen will display the correct syntax paradigm. [see below for common data retrieval commands.]

Command Mode Exercises

Select All From "table name" - This command will show you all the columns in a specified table. It is the most general of the select commands.

Select "column name" From "table name" - This command will show you only the columns you specify.

Sorted By "column name" - This command allows you to put the data you want to view in a particular order. The column name you specify will be put in numerical or alphabetical order.

Where -This command lets you add specific conditions to your selection.
A more detailed explanation is found below.

Command Mode Exercises

Data Retrieval Scenario

The town of Beni Mellal has two preparatory schools. Their identification numbers are 00137 and 00215. Some employees are registered at one school but work at the other. You want to know who those teachers are. Using RBase Command mode, retrieve the needed information.

Step_1. Which columns will you build keys for? Write them in the blanks below.

R> Build Key for _____ in Tpersinf

R> Build Key for _____ in Tpersinf

Step_2. Type in the commands on the above lines. In order to see if the commands were successful, you will have to look at the table. Key columns are marked by a [*]. If your screen did not show an error message, you can assume that the operation was successful.

Step_3. You want to use the **Select** command to retrieve specific data from the table **Tpersinf**.

Command Mode Exercises

First let's use the Select All command

- 3.1 Which table contains the data you want to see? Write the answer on the blank line below.

Select All from _____

The above sentence is the command that you should use if you want to view all the data in the table.

- 3.2 Type it in using **RBase Command mode**. Notice the data you view on the screen. Does it answer the question asked above? Why or why not?

Command Mode Exercises

Step 4. In order to get the information you want, you need a more specific command.

We know that if the school identification numbers in both columns are the same then the employee works in the school where he is registered.

If the identification numbers are different, then we know the teacher works in a school other than the one where he is registered.

4.1 You must arrange the two columns in a command relationship that will show you which columns are not the same. This can be done by using the **where** clause in a command. The **where** clause makes the command more specific. It allows you to view only data that meets the requirements of the **where** clause.

4.2 Look at the **where** clause operators below. Which one will allow you to display the data you want?

Command Mode Operator

Where Clause Operators To Compare Two Columns

Operator	Symbol	Definition
EQA	=A	Value of Column name 1 equals the value of column name 2.
NEA	<>A	Value of column name 1 does not equal the value of column name 2.
GTA	>A	Value of column name 1 is greater than the value of column name 2.
GEA	>=A	Value of column name 1 is greater or equal to the value of column name 2.
LTA	<A	Value of column name 1 is less than the value of column name 2
LEA	<=A	Value of column name 1 is less than or equal to the value of column name 2.

Command Mode Exercises

Step 5. Write your answer to the question in Step 4.2 on the blank line below

Step 6. Which columns in the table `tpersinf` contain the data you want?

Step 7. Fill in the blanks below with the column names and the operator you chose.

R>SELECT ALL FROM TPERSINF WHERE

Column Name	Operator	Column Name
-----	-----	-----

Type your command in:

Command Mode Exercises

Note: The screen should display column names and all the rows that meet the requirements of the **Where** clause. Because there are more columns in the table than the screen can show you at one time, you may not see the columns you want. It may be more helpful to view only columns that contain the data you want. For example, you may not be interested in the sex or salary of the employees but you are interested in the two identification numbers and the last name of the employee.

Step 8. Instead of the **Select All** command, you can use the **Select colname** command in order to view only the columns that you want.

Syntax for Select Column Name

```
R>Select "column name" from "table name"  
Where _____
```

Note: In order to select more than one column name, simply type in a comma [,] after each column name. [e.g. **Select colname1,colname2,colname3**]

Command Mode Exercises

- 8.1 Using the **Select Column Name** command, display the last name and school identification numbers of employees who do not work in the schools where they are registered. Only the three columns you asked for should be displayed on the screen.

12.6 Data Retrieval/ Space Saving

When you ask for certain data the computer will display it in the form of columns. The space for each column may be wider than you need. Two columns may occupy half a page. To avoid this you can use a space saving measure when writing your commands.

STEP 1. Using the Select colname command, display the last names and salary of teachers who work in school 00215.

The screen should display the two columns. You notice that there is a great deal of space between them. In order to avoid such wasted space, RBase allows you to control the amount of space that is allotted to each column.

Command Mode Exercises

Again, the syntax for this command follows the rhythm of other familiar **RBase** commands. Recalling the syntax for **Select colname** commands, this new command only requires that you specify the number of spaces you want to allot each column.

STEP 2. What are the columns you want to display?

Write them in the blank lines below. After each column name, type the number of space you want to allot to it.

Select _____, _____

STEP 3. In which table are the columns located?

Write it in the blank line after **From**:

From _____

STEP 4. What piece of data must equal 00215? Write

the name of the column which contains the data in the blank line after **Where**.

Where _____ = 00215

Command Mode Exercises

STEP 5. Type in your command. Notice the difference in space allocation. Now practice with other commands you can think of.

You now have an understanding of some of the basic questioning commands. The exercises above should give you a good idea of how to phrase your commands. Keeping what you have learned in mind, think of questions that you might need to ask of a personnel database, e.g. how much does a teacher in a particular school make, what subject does he teach etc. Once you have come up with a series of questions, write down on a piece of paper the commands that will display that information and begin experimenting with them on the computer. Feel free to ask the training assistants questions if you need help.

Command Mode Exercises

12.7 THE TALLY COMMAND

Obviously, when you are dealing with textual information the computer cannot add values in the column together. **R:Base** allows you to count the number of times an identical value exists for a column. For example, you want to know how many female teachers are working in a certain school. You can find this information by using the **Tally** command.

The **Tally** command counts the number of times a selected value occurs in a column. This command allows you to count text entries such as teacher, director or deputy librarian as well as numerical values.

12.7.1 Tallying Values

The syntax for the **Tally** command is as follows:

```
R>Tally "column name" From " table name"
```

To familiarize yourself with the command, find out how many males and females are included in your present database.

Command Mode Exercises

STEP 1. What column will tell you whether the employee is male or female?

Write your answer in the blank line where column name should go.

What table is the column in? Write your answer in the blank where table name should go.

R>Tally _____ From _____

STEP 2. Type your command in and the screen should display the number of males and females in the table.

Step 1 was a very simple operation and really did not tell you all that much. With the select command you can be more specific by using a "where" clause. You can use it with the tally command as well. The **where** clause operators you used with the **Select** command are used to compare two columns. The following operators are used when you are comparing a column to a value. They can be used in two formats, operator or symbols. You will notice that there are three operators that have no symbolic equivalent. They will be discussed shortly.

Command Mode Exercises

Operators to be used with the Tally command:

----- WHERE Clause Operators To Compare A Column to a Value

<i>Operator</i>	<i>Symbol</i>	<i>Definition</i>
EQ	=	Equal
NE	<>	Not equal
GT	>	Greater than
GE	>=	Greater than or equal to
LT	<	Less than
LE	<=	Less than or equal to
CONTAINS		Contains a text value
EXISTS		Contains data
FAILS		Contains a null value

Command Mode Exercises

The **Contains** operator allows you to look at textual data and select out any entry which contains the value you specify.

You will remember from the data you entered into your database that there are specific categories which contain no data. **R:Base** will assign that entry a zero. This fact, depending on the type of data you are working with, can sometimes be misleading. The **Exists** operator counts any entry in the rows whether it is an entered zero or just a lack of data. The **Fails** operator will count the rows where no data was entered. In other words, if you use the **Fails** operator, you can find out how many rows are empty as opposed to how many had zeros entered.

Command Mode Exercises

Tally with Where Clause.

The **Tally** command can be used with a where clause. In a previous example, you used the **Tally** command to find out how many male and female teachers there were in your database.

Now add a where clause to the same command. Find out how many male and female teachers there are in school 00215. Review the section on **Where** clauses above if you need to.

12.8 COMPUTING COLUMN TOTALS

You can retrieve column data using the **Select** command. An added feature allows you to display totals of individual columns. Of course, such a command can only be used with numerical data.

In section 12.5 you learned how to save space when displaying individual table columns. The syntax for computing column totals is exactly the same. Instead of typing in the number of allotted spaces, you type in the letter after the column you want to display.

Command Mode Exercises

STEP 1. Open the **Egypt** database.

STEP 2. Find out how many girls there are in the second grade. Use the syntax outlined below:

```
R>Select secongir=s from update2
```

STEP 3. The **s** that appears after **secongir** means that you want **R:Base** to total that column. If you want to total more than one column the syntax is the same. Following the same rhythm of the **Select** command, simply type in a comma [,] between each column name.

STEP 4. Experiment with the commands you have learned so far. Think about what data would be helpful in making decisions and then enter commands to display that data. Try to use the space saving and column total commands when possible.

Command Mode Exercises

12.9 LOGICAL OPERATIONS

In order to illustrate the logical operations in the R:Base package you will need to add Tables to your database.

Before you explore the logical, or relational functions, in R:Base look at the sample data collection form you have been given. Examine the table containing descriptive data, e.g. name of school, address, telephone number etc. Before you build a database table based on such a data collection table, can you suggest any changes in the data that the table contains ?

STEP 1. Open the database **TCAIRO**.

STEP 2. Follow the same instructions you found in **Section 5, Database Tables**. Build a table based on the school descriptive data. Make sure you begin by planning on paper.

Command Mode Exercises

12.10 THE UNION COMMAND

You now have two tables that contain very different data. The **UNION** command allows you to join tables together to form a new table. The new table you have just created contains at least one table column in common with **Primstud**. That column should be the school Identification number. To illustrate the **UNION** command we will use the following example.

Planner **x** is interested in viewing enrollment data in school **xx**. He also wants to view that school's address, telephone number and the name of the headmaster.

You are able to join your new table with **Primstud** to form a third table. The third table could then be the driving table for a specific report.

The syntax for the **UNION** command follows:

```
R>UNION table 1 With table 2 Forming table 3
```

Command Mode Exercises

- STEP 4.** Combine the two tables and name the third table **Schlstat** for School Statistics.
- STEP 5.** Look at the newly formed table in Definition Express. How many columns do you have now ?
- STEP 6.** Continue work on your database structure. Build a form for you new table and load some practice data. Practice the commands you have learned so far. How can you tailor the work you have done so far to your own work needs ? Begin work on a database structure that will aid you in your work situation.
- STEP 7.** There are many **R:Base** commands that you have not learned yet. Ask your instructors or consult the **QUE R:Base** manual if you want to learn more logical functions or other commands that will help you in your own work situation.