SUPERBASE 64

USER'S MANUAL

PRECISION SOFTWARE

1983

S U P E R B A S E 64

INTRODUCTION

This manual is in three main parts. The **TUTORIAL SECTION** aims to train you step by step in how to use the major options of Superbase. The **REFERENCE SECTION** is for use as an 'Encyclopedia' covering all of the options available from the Menus. The **PROGRAMMING SECTION** is for more advanced users who want to set up Programs to carry out long sequences of Superbase operations automatically.

As well as being an easy to use Database System controlled by Menus from which you select the options you want, Superbase is also a powerful Applications Generator and Database Programming Language. This allows you to automate the operations you will most frequently require and even to set up your own User Defined Menus to tailor the system totally to your own needs.

These features go beyond the comprehensive self-contained menu system, which is perfectly adequate for day to day operations. The **PROGRAMMING SECTION** covers the more advanced aspects of the system.

We strongly recommended that you first work through the **TUTORIAL SECTION** which assumes no previous knowledge of either computers or Database Systems, and use the **REFERENCE SECTION** as and when you need it for further detail.

Once you feel that you have mastered Superbase's various commands and options you may then decide to read through the section on programming Superbase to extend your power over the system.

A number of **APPENDICES** are also provided with information on how to set Superbase for the RS232 printer interface, the meaning of the various **ERROR MESSAGES**, a **TECHNICAL APPENDIX** containing Superbase parameters, a **SELECTED GLOSSARY** of terms, and an **ALPHABETICAL INDEX**.

A Registration Certificate follows this introduction. It must be completed and returned to Precision Software Limited so that you qualify for future modifications and updates to Superbase.

Proprietary Rights

Limited Warranty

The contents of this manual are subject to change without notice and do not represent any commitment on the part of Precision Software Limited.

No warranty is made with respect to the software described herein. The sole obligation of Precision Software Limited shall be to make available modifications and updates in accordance with a policy of continuous improvement, and subject to its normal trading terms and conditions.

Precision Software shall have no liability or responsibility to purchaser or any other person or entity with respect to any liability, loss or damage caused or alleged to be caused directly or indirectly by this product, including but not limited to any interruption of service, loss of business or anticipatory profits or consequential damages resulting from the use or operation of this product.

The above is a limited warranty and the only warranty made by Precision Software Limited. Any and all warranties for merchantability and/or fitness for a particular purpose are expressly excluded.

Acknowledgements

CBM is a trademark of Commodore Business Machines Inc.
Diablo is a trademark of the Xerox Corp.
Qume is a trademark of Qume.

This manual was created on Easy Script by Charles Langley, edited using Superscript II by Dr Bruce Hunt and printed on a Diablo 630 printer, with a 12 pitch letter gothic daisy wheel. The Manual was checked using Superspell II yielding the following statistics:

Total number of words	=	49693
Number of unique words	=	2399
Number of sentences	=	2733
Number of paragraphs	=	2700
Average word length	=	4.1

First published August 1983.

PRECISION SOFTWARE AND SUPERBASE

Superbase is just one of Precision Software's range of systems for Commodore computers. Superbase was conceived by Simon Tranmer, the designer who created the outstanding Superscript word processor and Superspell spelling checker for the Commodore 4000 and 8000 series (Easy Script and Easy Spell on the Commodore 64). Simon also enhanced Superscript II for the Commodore 700/B series and was joined by co-designer Tom Cranstoun for the Superbase project.

Superbase is available on the Commodore 64, the Commodore 8096 and the Commodore 700/B computers. Superoffice, a system that integrates full word processing with advanced database management, is available on the Commodore 8096 computer.

THE SUPERBASE 64 PRODUCT

The Superbase 64 product consists of a manual and 2 disks held in a pocket inside the front cover. The two disks are identical, and one should be kept in reserve as a backup in case of accidents. A replacement disk can be obtained from Precision Software on return of the damaged disk. There is a replacement charge of £5 or the currency equivalent (plus overseas mail) per diskette.

TUTORIAL CONTENTS

TUTORIAL CONTENTS

TUTORIAL CONTENTS

REFERENCE SECTION CONTENTS

REFERENCE SECTION CONTENTS

REFERENCE SECTION CONTENTS

REFERENCE SECTION CONTENTS

SUPERBASE 64

REGISTRATION CERTIFICATE

Precision Software Limited has a policy of reviewing its products constantly. As part of this policy we provide modifications and updates to our Registered Users free, except for a small handling charge to cover media, postage and packing.

To become a Registered User of Superbase 64 and to qualify for modifications and updates, please complete this Certificate and return it to:

Precision Software Limited
4 Park Terrace
Worcester Park
Surrey KT4 7JZ
England

REGISTRATION CERTIFICATE

Name Telephone

Company

Address

Zip/Postcode

Supplier

Date of Purchase

Intended Use/Application

Send details of the Superbase 64 Application Program Prize - Yes/No

Tear along the dotted line and return to:

Precision Software Limited, 4 Park Terrace, Worcester Park, Surrey KT4 7JZ England

PROGRAMMING SECTION CONTENTS

PROGRAMMING SECTION CONTENTS

TUTORIAL ONE

1.1 INTRODUCTION

Many owners of Superbase will not have used a Database Management System before. Some will have bought their CBM 64 especially for using Superbase and may not have had any experience of computers.

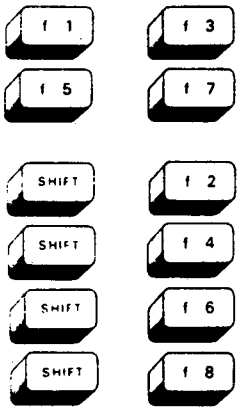
This first tutorial therefore aims to be a very basic introduction to the most commonly used features of the program. However, even at this elementary level the worked examples are designed to provide you with a useful application at the end.

After a general introduction to the keyboard of your CBM 64 you will be guided step by step in setting up an Address Book file which you may keep and use to store names, addresses, phone numbers and other information about your friends and business contacts.

If you decide that you want to finish before the end, then read section 1.8 'End of Session' before switching off your computer or disk drive.

1.1.1 THE KEYBOARD

The keyboard of the CBM 64 consists of two main parts: the **MAIN KEYBOARD** in the center and the **FUNCTION KEYS** on the right. Sit in front of the computer now, and make sure that you can find all of the keys mentioned below.



There are four **FUNCTION KEYS**, each of which has a number on the top and a number on the front. The numbers are from f1 through to f8.

The odd numbers, f1, f3, f5 and f7 are obtained simply by pressing the function key with the required number on top. The even numbers, f2, f4, f6 and f8 are obtained by holding down the SHIFT key at the same time as pressing the function key with the required number on the front.

To simplify we shall refer to the key with f1 on top, pressed on its own as 'the f1 key' and the same key pressed while the SHIFT key is held down as 'the f2' key. This notation is also used for the key pictures which accompany the text.

These function keys are the main control keys used in Superbase for selecting the various options that are available.

The main keyboard contains many keys used, among other things, to type words and numbers into Superbase.

Some keys have two symbols on top, such as the '1' key at the top left which has an exclamation mark (!) above the number one.

Pressing these keys on their own will result in the lower of the two symbols appearing on the screen. To obtain the higher of the two symbols (such as the exclamation mark), hold down the SHIFT key while you press the key.

If you hold down the SHIFT key and press one of the keys with a letter on the top you will obtain capital letters. Pressing these keys on their own will give you lower case.

GRAPHIC SYMBOLS The Graphic symbols on the front of the keys are not used very much in Superbase, except for drawing borders and for underlining. Ignore them for now until they are mentioned later on.

SHIFT We have spoken of the SHIFT key, but in fact there are two of them, one on the bottom left and one on the bottom right. These two SHIFT keys have exactly the same function and there are two of them to assist you while typing.

SHIFT/LOCK With SHIFT on Menu commands are inoperative.

RETURN Another important key in the main keyboard is the RETURN key on the right hand side.

This key is used very frequently in most programs, including Superbase, and usually signifies that you have finished typing and want the program to respond. The RETURN key is also used in Superbase for other purposes but these will be explained later as they arise.

CTRL The key on the left hand side of the keyboard marked 'CTRL' is called the CONTROL key. Its only function in Superbase is with the '1', '2' and '3' keys for changing the color of the typeface, screen and screen border, and with 'p' for printing screens.

RUN/STOP Just below the 'CTRL' key is a key marked 'RUN/STOP'. This key is used, together with the 'f1' key, to terminate a Superbase command.

CURSOR MOVEMENT

Two more keys which you will be using a lot are the CURSOR CONTROL keys. These are at the bottom right of the keyboard and are marked 'CRSR' and have arrows on them.

When you type onto the screen you will notice a flashing square just where you are about to type. This is known as the CURSOR and it can be moved around the screen with the keys marked 'CRSR'. This

is how you control where on the screen your typed information appears.

The two cursor control keys are distinguished by the direction of the arrows displayed on them.

The left hand cursor control key has vertical arrows, and is used for moving the cursor up and down.

To move the cursor down simply press the key on its own. If you hold the key down it will repeat and so will take the cursor to the bottom of the screen.

If you want the cursor to move upwards then you must hold down the SHIFT key while pressing the vertical cursor key. Again you may press the key once to move up one line or else you may hold it down to repeat the upwards movement.

Similarly, the key marked 'CRSR' with horizontal arrows can be used on its own for moving the cursor to the right, or with the SHIFT key held down for moving the cursor back to the left.

Again the key will repeat if it is held down or will only move the cursor one space if it is pressed once.

HOME AND CLEAR

The HOME key at the top right is also a cursor control key. It moves the cursor directly to the top left corner of the screen (known as 'Home').

Pressing this key with the SHIFT key held down clears the screen of everything you have typed. The 'CLR' stands for 'CLEAR'.

INSERT AND DELETE

Next to the CLR/HOME key is the INSERT/DELETE key, used for deleting text and for inserting characters into the text.

To delete text, place the cursor just in front of what is to be deleted and press the key marked INST/DEL. You will see everything to the right of the cursor move back to cover the unwanted characters.

To insert text place the cursor on the character before which you want the inserted text to go. Hold down the SHIFT key and press the INST/DEL key and you will see a gap open up and everything to the right of the cursor, as well as what is immediately underneath it, move along to the right.

Continue to press the INST/DEL key until the gap is large enough and then you may let go of the SHIFT key and type whatever text you wanted inserted.

As you have probably discovered, the long bar at the bottom of the

main keyboard is the SPACE BAR and is for typing blank spaces.

1.1.2 STARTING UP

Before starting to use Superbase, you must create a data disk to hold your database. The Superbase disk itself is always removed from the disk drive once the program is loaded, and cannot have databases created on it. Therefore if you want to use the Superbase HELP screens, you must copy them off the Superbase disk onto a disk of your own. This applies also to the demonstration database used in the Tutorials and the set-up program, 'start', which is described further in the Technical Appendix.

INSERT Superbase DISK

Since the Superbase disk is protected and cannot be copied using normal copying procedures, you need to run a special program for creating data disks. This is supplied on the Superbase disk. After switching on the computer system as described in the Commodore 64 manual, place the Superbase disk in the disk drive (this will be drive 0 if you are using a dual drive unit). Close the drive door. Type the command:

```
load "sb",8,1
```

and press RETURN. Use the INST/DEL key to correct typing errors. Wait while Superbase is loaded. This will take about 2 minutes on a 1541 disk drive, and usually less on a 4040 disk drive. When Superbase is loaded you will see the following message:

```
Remove Program Disk
Insert Data Disk and Press Return
or
Press f1 to Create Data Disk
```

FORMAT NEW DISK

Normally you will place an existing data disk in the drive and press RETURN, causing 'start' to run automatically. Now, however, you need to create your first data disk, so press the f1 function key to the right of the main keyboard. You will see the following message:

```
Insert Blank Disk in Drive 0
Press Return to Continue
```

Even if you are using a dual drive unit, this operation requires you to use drive 0. Place a disk (it can be either a new disk from the box or an old disk that you wish to re-use) in drive 0 and close the drive door. Press RETURN. Next you will see this message:

```
All Data on Drive 0 Will Be Destroyed
```

Are You Sure?

This helps to ensure that if you are using an old disk you definitely do not need any of the information on it. When you are ready to proceed, type 'y' for 'yes'. Next this prompt appears:

Enter Disk Name,id

You must give your data disk a name and an identification code. Use any name up to 16 characters, such as 'work disk' (use any characters including spaces, but excluding the colon ':'). Then you will need a 2 character identifier - it must be 2 exactly. Use 'aa' or some such code (vary it to 'ab' etc. as you require more disks). Type the disk name and code in now, including the comma between them. Press RETURN, and your disk will be prepared for use.

COPY FROM SUPERBASE DISK

The next job is to copy the Superbase HELP screens and the demonstration database 'Training' from the Superbase disk to your new disk. This is done in a couple of easy steps. When the flashing message 'Insert Source Disk' appears on the screen, remove the new disk and replace the Superbase disk in the drive. Press RETURN. You will see the message 'Please Wait', followed by a series of dots which appear as Superbase loads up the data it has to copy.

When Superbase is ready, the message 'Insert Destination Disk' will flash on the screen. Remove the Superbase disk and replace the new disk. Press RETURN. Again you will see the message 'Please Wait' and the row of dots as before. Superbase is copying the necessary files onto the new disk.

Depending on the size of the HELP screens and demonstration database that accompany your version of Superbase, more than one copying cycle may be needed. If the message 'Insert Source Disk' appears again, go through the actions described above, starting at the sentence 'When the flashing message....' When the copy is completed, Superbase will close itself down and return the computer to the state it was in just after you switched it on. If you are using an IEEE cartridge such as the DAMS cartridge, you must switch the computer off and then on again.

To carry on, re-insert the Superbase disk and type the command:

load "sb",8,1

and press RETURN, as you did when you first started. Read through to the next section and carry on with the tutorial.

There is another way of creating data disks with HELP screens. Once you have learned how to reach the BACKUP Option on the Maintain Menu, you can duplicate an existing disk with Superbase's own BACKUP facilities. (You may wish to remove the 'Training' database when it is no longer needed. You can always get it back from the original Superbase disk.) You have in fact already used the BACKUP procedure to create this first data disk.

INSERT DATA DISK and SELECT DATABASE

When Superbase has loaded, take the Superbase disk out of the Drive Unit and put it safely back into its jacket. Now insert the Data Disk into the Drive Unit and press RETURN. The set-up program 'start' will run automatically. When asked for the name of a database, type 'Training' and press RETURN.

If you ever need to change disks because you put the wrong one in first, always use the 'database' command after inserting the new disk, to reset the system. Keep your backup disks for backup, not work.

You will see a list of files and a number of dotted lines and you will be asked 'Enter Filename'.

A Database is a collection of up to fifteen files and you can have as many databases as you can fit on your disks. Those files you see listed are provided for later sections of the tutorial

Right now we are going to create a new file called 'Addresses'. Type 'Addresses' and press RETURN. Superbase will respond by telling you that this file does not as yet exist and will ask if you want to create it.

Type 'Y' for yes and the Disk Drive will whir for a few seconds and the screen will clear.

The words 'Mode: Format' will appear in the message area at the top of the screen, telling you which Superbase OPERATION is in progress.

1.2 CREATING THE ADDRESS BOOK

The blank screen in front of you is the empty page on which you are to draw the screen layout of the Address Book.

The message 'Mode: Format' tells you which of the many options of Superbase you are currently in. The FORMAT option is used for setting up the screen layouts of the files of information to be held in Superbase. With this option you determine exactly how the information in the file is to be displayed on the screen.

Whenever you select a file that does not yet exist you will be asked whether you want to create it. If you reply 'y', you will be automatically put into the FORMAT Option.

There are two types of entry in a screen layout: the descriptive text, in the present case the names of items such as 'Name', 'Address' and so on; and the FIELDS, as they are called, where such items of information such as 'Jones' and '16 Normandy Rd.' are to be held in the record.

First of all we shall enter the names of the various items of information we are going to store in the Address Book records.

1.2.1 ENTERING THE FIELD NAMES

Using the cursor control keys, move the cursor down four lines and along three spaces.

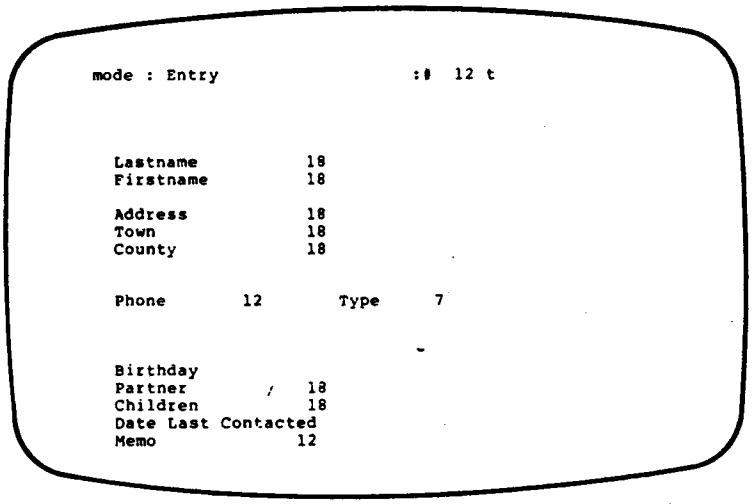
Hold down the SHIFT key to obtain a capital letter and type 'Lastname'.

If you press the RETURN key at this point you will find that the cursor moves down to the start of the next line.

Position the cursor immediately below the 'L' of 'Lastname' and type 'Firstname' (the reason for not leaving spaces between these words will be explained later). Use the cursor-down key to skip a line, position the cursor below the 'F' and type 'Address'.

Superbase does not care whether you use capitals or small, either when setting field names or when referring to them later.

Using the screen diagram below, continue typing the various headings, leaving blank lines as shown. Again, use the cursor control keys to move around the screen. The numbers in the diagram below do not appear on your screen: they are the field lengths, shown here for you to refer to later.



When you have finished, use the 'HOME' key to take the cursor back to the top of the screen.

1.2.2 SETTING THE FIELDS

You have just typed in what are known as the 'Field Names' for your Address Book records. Fields are blank slots where the various items of information are held in the records. A 'record' in an Address Book file will be like a single page holding the details of one particular person.

Having entered the names of the fields we now have to set the fields themselves.

THE RECORD KEY

Place the cursor six spaces along from the end of 'Lastname' and press the 'f1' key. You will see the word 'Mode' flashing in the message area at the top of the screen. Now press 'K' for 'key.'

The message 'Set Key' will be displayed and there will be a small rectangular marker on the screen just where you had the cursor positioned.

The small marker is the field-start marker. It shows you where the 'slot' into which the last names are to be typed in your records begins.

'Key' is the name of the type of field you are setting. Every record must have a 'Key Field', as this is the field that Superbase uses to store the records in alphabetical order in the file.

In this case we are going to use the last names to order the records, just as you might do in an ordinary Address Book where you would enter the names and addresses alphabetically according to last name.

COUNTING THE LENGTH OF THE FIELD

You will also see the number '1' in the right hand side of the message area. Use the cursor-right key to move slowly along the line and observe that the number in the right hand message area increases as you move along. This number is the length of the field you are setting, the field-length indicator. That is to say, it is the maximum number of characters that will be required in the 'Lastname' field.

When this number reaches '18', press the RETURN key. You will see a striped square appear. This is the field-end marker, and tells you where the 'slot' for the last name in your records is to end.

THE FIRST TEXT FIELD

Next we shall set the field which is to contain the first name. Place the cursor immediately below the last name field-start marker, press the 'f1' key and n press 'T' for 'text'. This time

the message displayed will be 'Set Text'.

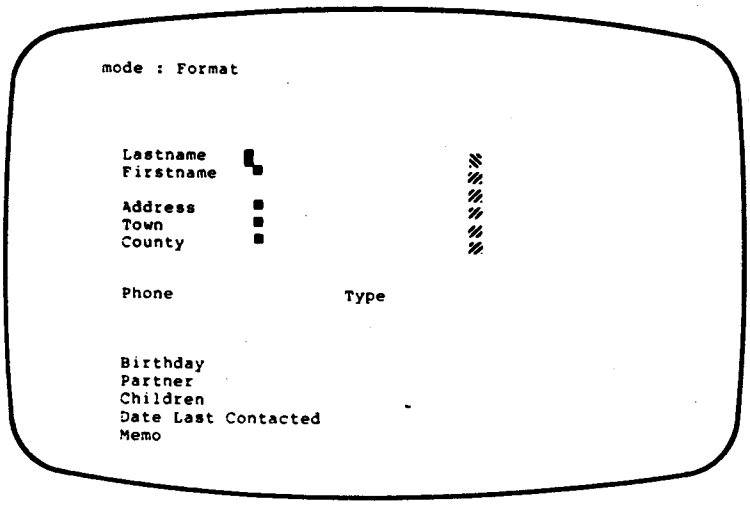
A text field is for containing words or combinations of words and numbers. Move the cursor along until the field-length indicator (the number in the top right message area) is again '18', then press RETURN. Once more you will see the field-end marker appear, this time with its stripes going the other way.

MORE TEXT FIELDS

Now you are getting confident, you may make a mistake, such as pressing RETURN too early. If you do, position the cursor on either of the field markers and press 'f1' followed by 'E' to erase the marker. Then start again.

We need three more text fields for 'Address', 'Town' and 'County', each of which should be the same length as the 'Firstname' field. Set these fields in the same way, positioning them exactly below the 'Firstname' field.

The screen diagram below represents how your screen should appear at this point.



Now we want to set the 'Phone' and 'Type' fields. These will again be text fields but will be shorter than the text fields we have set so far. Position the cursor one space along from the end of the word 'Phone' and press the 'f1' key followed by 'T' for 'text'. Move the cursor along until the field-length indicator says '12'

and then press RETURN.

Now move the cursor along to one space after the word 'Type' and press 'f1' again, followed once more by 'T'. Move the cursor along until the indicator says '7' and press RETURN.

You should now be able to set text fields quite easily, so it is time we tried a field of another type.

A DATE FIELD

Move the cursor to the same line as the word 'Birthday' and immediately below the name and address fields. Press the 'f1' key as before, but then press 'D' instead of 'T'.

You will see the message "Set Date" appear and the cursor will automatically position itself seven spaces along to make room for the date field. Press RETURN.

REMAINING FIELDS

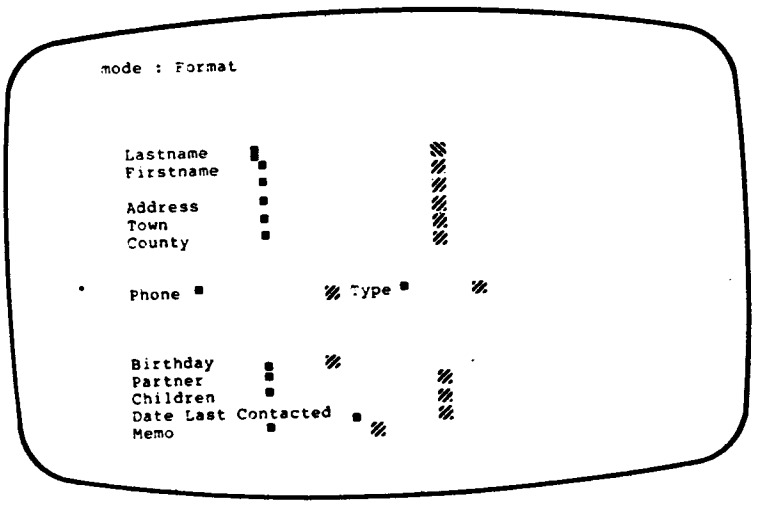
Place the cursor below the date field-start marker ready to set another text field to contain details about the person's partner.

Set this field in the same way as you did for the other text fields above, and then position the cursor on the next line down to set the field to contain details about children.

After setting the 'Children' field, we shall create another date field for storing the date the person was last contacted. Place the cursor two spaces along from the word 'Contacted' and set the date field by pressing first the 'f1' key and then 'D'.

Finally, we shall set one further text field called 'Memo', which is to contain the name of an optional screen of extra notes relating to the person in the record. Place the cursor one space along from the word 'memo' and press 'f1' followed by 'T'. Move the cursor along twelve spaces and press RETURN.

This is what your screen should now look like:



1.2.3 INVERTING AND COLORING THE SCREEN LAYOUT

We could store the screen layout as it is now, but first we shall embellish it a little.

Hold down the key marked 'CTRL' (the Control key) at the top left of the keyboard and press each of the keys marked '1', '2' and '3' several times.

You will notice that the first of these keys changes the color of the characters on your screen, the second changes the color of the screen itself and the third changes the color of the border around the screen.

Experiment a bit until you get a combination of colors you find pleasing. Superbase will remember that the Address Book file is to be displayed in the colors you have chosen.

Each of your files can be a different color, helping you to identify it at a glance.

Another way we can improve the appearance of the Address Book file is to invert a line of the screen and to invert the whole screen itself. Place the cursor anywhere on the same line as the word 'Lastname' and press the 'f1' key followed by the 'I' key ('I' stands for 'Invert').

The result is that this line now stands out prominently.

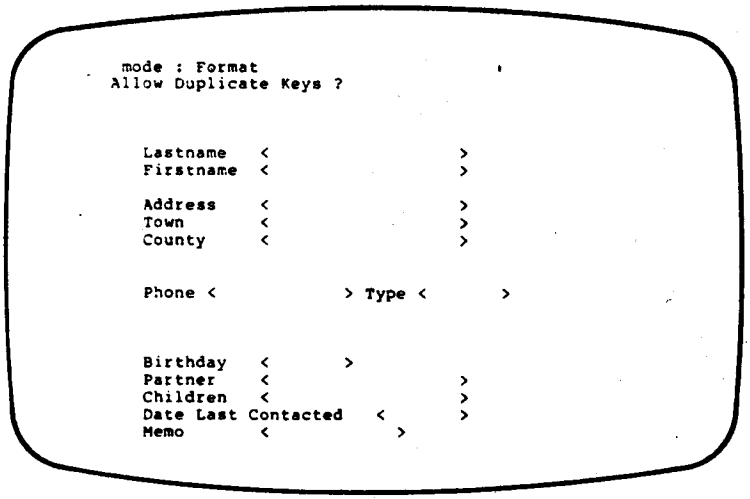
Now press the 'f1' key again this time followed by the 'S' key ('S' stands for 'Screen'). You should see the whole screen invert its colors, leaving the line we inverted highlighted from the rest.

This completes the design of the Address Book file.

1.2.4 STORING THE SCREEN LAYOUT

To signify to Superbase that you have finished creating the Screen Layout and want to store it, press the 'f1' key followed by the 'RUN/STOP' key.

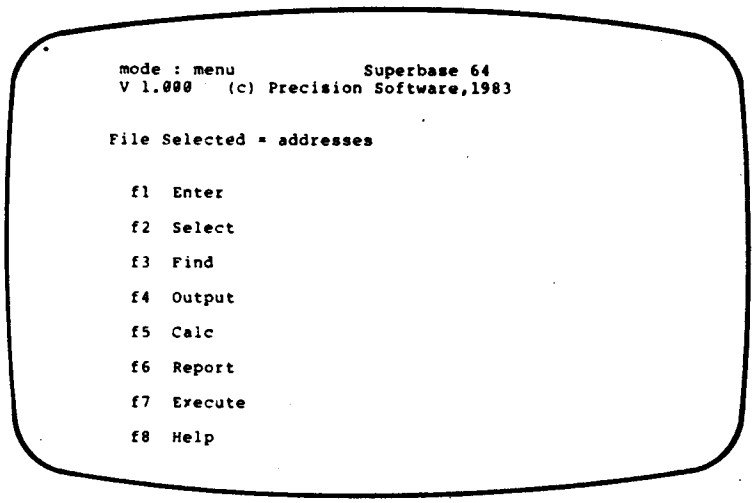
The message 'Finished' will be displayed at the top of the screen and you should see the field start and end markers replaced one by one with angle brackets as in the diagram below:



You will be asked 'Duplicate Keys Allowed?'. Duplicate keys will be explained in the next tutorial. For now type 'N' for 'No' and you will see the Main Menu appear on the screen.

1.3 THE MAIN MENUS - MENU 1 AND MENU 2

The MAIN MENU, or as it is often called, MENU 1, is your major means of access to the most commonly used options of Superbase. It consists of a list of eight options together with the control keys used to obtain them.



Menu 1 also provides you with a COMMAND LINE at the top of the screen onto which you can type a string of commands directly into Superbase. For instance, instead of pressing 'f1' to obtain the ENTER option you could instead type 'enter' onto the command line.

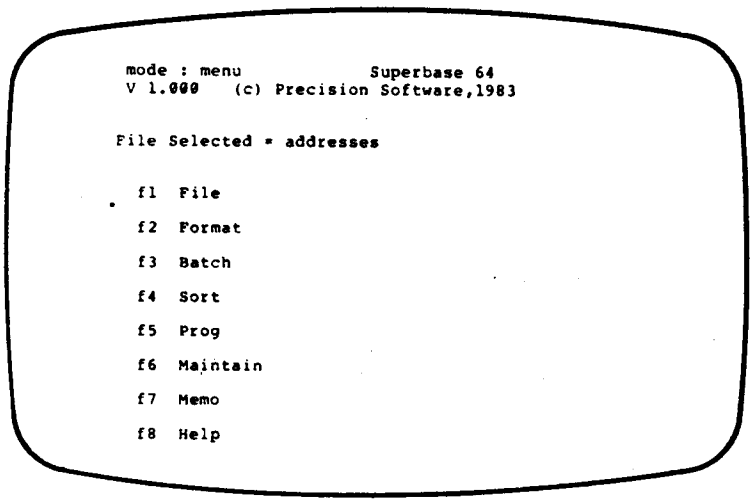
This means that Superbase has a dual control system: it is both menu-driven and command-driven.

The advantage of using the command-driven method of control is that whole sequences of commands can be entered at one go. This greatly increases the power of Superbase as you will see in the second and third tutorials.

During this tutorial, however, we shall restrict ourselves to using the menu-driven method of control and suggest that users continue to do so until they have more experience of the various features of Superbase.

Superbase in fact has two Main Menus, MENU 1 and MENU 2. To obtain MENU 2 from the MENU 1 or MENU 1 from MENU 2, just press RETURN.

Try it now and you will see the list of options below on your screen:



Notice that the second option listed on Menu 2 is the FORMAT Option which you used just now. Obtaining the FORMAT Option from this menu enables you to revise the screen layout of the current file.

You are automatically put into the FORMAT Option when you are creating a new file, either on start-up if you choose a file that does not exist (as you did in the previous section), or if you create a new file by means of the FILE Option on Menu 2.

Press RETURN once more and you will be back at Menu 1.

1.4 ENTERING INFORMATION INTO THE ADDRESS BOOK

From the Main Menu, press the 'f1' key to obtain the ENTER option.

You will see the Address Book screen again and the cursor will be positioned in the NAME field at the top of the screen.

KEY

Type a name, say 'Dripp' and press RETURN. The cursor will now be in the second field.

TEXT

Now type a first name, say 'John', and press RETURN again. Continue in this way, either typing in information or else just pressing RETURN if you want the field to remain empty, until you come to the 'TYPE' field into which you should type 'plumber'.

DATE

The next field is the one we have given the name 'BIRTHDAY'. Remember that we set the 'BIRTHDAY' field up as a date field. Date fields require dates of the form '12jan83' or 'oct1882' (the month can be either capitals, lower case or a mixture of the two).

Enter a date in this way and press RETURN. Notice that Superbase calculates the day of the week and displays it at the top of the screen and automatically converts months to uppercase.

REMAINING FIELDS

Continue entering information into the fields until you come to the field called MEMO. Enter into it the words 'John Dripp' and press RETURN.

STORE NEW RECORD

You will see the message 'Press Return to Store' displayed at the top of the screen. Press RETURN once more and the record will be filed.

This 'Press Return to Store' message is always displayed after you have pressed RETURN from the last field (whether or not you have typed information into it). You can obtain it at any time during the ENTER option after you have left the key field by holding down the SHIFT key and pressing RETURN.

You may want to do this if you want to leave some of the later fields empty, but remember that the key field is always a FORCED FIELD, which means that you must enter some information into it in every record.

As it happens, any field can be set up as a Forced Field during the formatting process, but we declined to do this for any of the fields in the Address Book file.

OPTION TO ADD MORE RECORDS

After filing the record you have just entered, Superbase will display the message: 'Press Space to Enter Another'. If you press RETURN at this point you will be returned to Menu 1 from where you can select another option. If instead you want to enter another record then press the space bar and the screen will display another blank record with the cursor in the first field ready for information to be entered.

Enter several more records since they will be needed later when we come to examine the SELECT Option. Make sure every key is unique, you can add a number to a name to ensure that this is so.

1.5 CREATING A MEMO SCREEN

You may have been wondering why a field was included in the Address Book file called MEMO.

Independently of the files of records, Superbase allows the creation of whole screens of text called 'Memo Screens'. The purpose of these screens, which can be called up at any time from the Menu 2, is to enable you to write notes to yourself or to other users of Superbase which are stored and easily retrieved.

Sometimes in an ordinary address book you may wish to include special notes about one of the people listed that will not fit on the page. Similarly, since the records in our electronic address book have a pre-defined format there may be information you wish to include that cannot be fitted on the record.

This is where the Memo Screens come in handy. Suppose you have a record such as the one we entered earlier relating to a plumber called Mr Dripp, and you want to make a note of the various rates he charges and of the work he has done for you before.

SELECT MEMO

Use the RETURN key to obtain Menu 2 and from it select the MEMO Option by pressing the 'f7' key. You will see a blank screen with the words 'Enter Memo Name' at the top of the screen.

Type 'Dripp' and press RETURN (a Memo screen can have any name of up to sixteen characters). You will see the words 'Mode: Memo Writer' above the screen with the cursor in the HOME position.

WRITE MEMO

You can now type anything you like onto the screen and when you are finished press the 'f1' key followed by the 'RUN/STOP' key to store the Memo. You can recall the Memo screen at any time from Menu 2 by selecting the MEMO Option and typing 'Dripp' when asked 'Enter Name'.

RECALL MEMO

Try recalling the Memo screen in this way to prove to yourself that it is there, and when you are finished with it press the 'f1' key followed by the 'Q' (for Quit) key to return to Menu 1.

A Memo Screen can be revised by recalling it by using the 'MEMO' command from the Main Menu, typing your amendments, and then re-storing it by using 'f1' followed by 'RUN/STOP' rather than the usual 'f1' followed by 'Q'.

1.6 THE HELP SCREENS

There is a special kind of Memo Screen called a HELP SCREEN which can be obtained from either of the Main Menus by pressing the 'f8' key.

A number of these are already built into the system and are for providing advice about the various options provided by Superbase. If you need such advice, select the HELP Option and when asked 'Enter Which Help Required', type the name of the option you want help with.

Later on you may want to redesign one of the Help Screens or to create a Help Screen for one of the sub-options for which there is no built-in Help Screen available.

This can be easily done by means of the MEMO Option. All you have to do is create a Memo Screen as normal, but prefix its name with an 'h'. For example, if the screen is to contain advice about the Invert Screen and Invert Line commands you encountered above, then call the Memo Screen you create 'hinvert'.

Having done this you can then obtain this advice by selecting the HELP Option and typing 'Invert' when asked 'Enter which help required'.

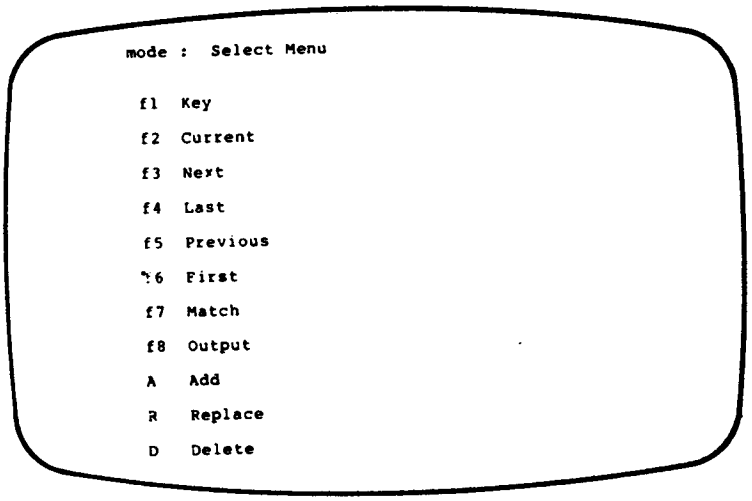
1.7 SELECTING RECORDS

In the section to follow you will be shown how to use one of Superbase's most important and frequently used options, the SELECT Option.

If you haven't already done so, make sure that you have entered at least three records into your Address Book file in addition to the record you entered for John Dripp the plumber in section 1.4 above.

From Menu 1 choose the SELECT Option by pressing the 'f2' key (remember, this is the 'f1' key with the SHIFT key held down).

You will see another menu on the screen. This is the SELECT MENU and it provides you with a choice of commands for obtaining the various records in your file.



1.7.1 BROWSING THROUGH THE RECORDS

Press the 'f6' key to execute the 'First' command. You will see a record from the Address Book displayed on the screen. This will be the record belonging to whoever has the name that comes first alphabetically.

You will also see a list of letters at the top of the screen. These are abbreviations for the commands you saw a moment ago on the SELECT Menu. If you can't remember what the commands were, then press RETURN to obtain the full menu again.

You can obtain the full SELECT Menu in this way at any time you are in the SELECT Option. Press 'f6' again to obtain the first record in the file and then press 'N' which is the abbreviation for the 'NEXT' command.

You will see that by repeatedly pressing 'N' you can step through all of the records until you get to the end of the file.

Next try the 'P' key which abbreviates the 'PREVIOUS' command. This is just the reverse of the 'NEXT' command in that it allows you to step backwards through the file.

1.7.2 SEARCHING FOR A PARTICULAR RECORD BY KEY

Press RETURN to obtain the SELECT Menu and then press RETURN again. You will find that you are now back at Menu 1. This is how you exit from the SELECT Option when you have finished using it.

Select the SELECT Option again,but this time choose the 'KEY' command from the SELECT Menu by pressing 'f1'. This is one of the commands you can use to quickly find a record in the file.

You will be asked 'Enter Key'. The 'KEY' in any record is whatever you entered into the 'KEY FIELD' for that record (see above, section 1.2.2).

Type 'Dripp' and press RETURN. Superbase will find and display the record belonging to John Dripp the family plumber.

Press 'K' and you will be asked for another key. Choose the surname of somebody else you have put into the Address Book and press RETURN and you will see the record you have selected appear on the screen.

PARTIAL KEYS

Superbase always looks for the alphabetically closest key. If a key is found that starts with what you typed in but has more characters, then that record will be selected and the message 'Partial Match' will appear at the top of the screen. If there were no leading characters in common with the specified key, the message would be 'Key Not Found' and the record selected would be the next record in the file.

1.7.3 SEARCHING FOR RECORDS WITH THE 'MATCH' COMMAND

Suppose you wanted to find John Dripp's record but were unable to remember his name. Press the 'M' key to execute the 'MATCH' command and you will be presented with a blank record with the cursor in the first field and the request 'Select Match Data' at the top of the screen.

ENTERING DATA TO BE SEARCHED FOR

Using the RETURN key or the cursor-up and cursor-down keys you may move the cursor to any field which contains the information you remember. For example, suppose you remembered that the man you wanted was a plumber. Move the cursor down to the 'TYPE' field and type 'Plumber'. Superbase will accept any mixture of upper and lower case letters for a search. Thus 'PLUMBER', 'plumber', and 'plumBER' are all the same as far as matching is concerned.

If you finished here the MATCH command would find you all of the plumbers in your address book. You could narrow down the search by specifying more information you remember, such as that his first name was 'John' for instance. When you have finished entering your Match Data, hold down the SHIFT key and press RETURN (or move to the last field and then press return).

Once the first such record has been found you could press 'M' again to obtain the next record which satisfies your match criteria.

LOOKING AT NEIGHBORING RECORDS

You could also use the 'N' for 'NEXT' and 'P' for 'PREVIOUS' commands at this point to browse through neighboring records and then use 'M' again to return to the next matching record from where you left off.

MATCHING ON CHARACTER PATTERNS

There are other kinds of criteria you can specify when entering Match Data, such as 'Jo*' which will find records with 'Jones', 'Johnson', ''Jonah' and so on in that field, or 'Wi???s' which would find 'Wilkes' and 'Willis' but not 'Wilkinson' or 'Williams'.

This is called **'pattern matching'**.

ALTERNATIVE PATTERNS IN THE SAME FIELD

You could also use the '/' character to specify 'Plumber/Builder'. This would find any record with EITHER 'Plumber' OR 'Builder' in the specified field.

The MATCH command is an extremely powerful and flexible facility. For further details of these and other types of Match Criteria see the Reference Section 6.2.7.

1.7.4 REPLACING A RECORD

We have just looked at ways of summoning up records onto the screen. Now we shall examine how you can change the details in a record once you have found it.

Use any of the commands you learned how to use above to obtain one of your records on the screen.

When you have done this press 'R' for 'REPLACE' and you will see the cursor in the second field.

You can move the cursor around the record by using the RETURN key or the cursor-up and cursor-down keys, but you will find that you cannot enter the Key Field.

The reason for this is that you are editing an existing record rather than adding a new record, so the key must remain the same for Superbase to recognize it as the record you selected.

By typing over the information in the fields, make various changes to the record and then by using either SHIFT/RETURN or RETURN from the last field, obtain the 'Return to file' message. Press RETURN again and store the modified record.

By using the 'FIRST' and 'NEXT' commands satisfy yourself that the old record has indeed been replaced in the file.

1.8 END OF SESSION

We have now covered enough for one tutorial. You have seen how to set up a file, enter information into it and search through the file for records to view or to replace.

If you want to finish for now you should return to either of the Main Menus and type 'quit' on the command line, followed by RETURN.

This is the recommended way to close down Superbase. Your files and records are always maintained safely and fully up-to-date, so you can in an emergency just remove your data disk from its drive, provided that (1) there is no physical movement of the read/write heads going on, and (2) that you are at either of the Main Menus, and switch the system off.

HINTS AND TIPS

You are strongly advised to BACK-UP your data disk regularly. A particularly important time to do BACK-UP is before any major system update to ensure the integrity of your data. BACK-UP is done from the MAINTAIN Option. See Reference Section 15.4.

On changing data disks always ensure that you reselect your database by typing the command 'database' on the command line and following the prompts.

Do not duplicate the name of a database. If you use the name of an existing list or file, it will be overwritten by the new list.

LABELS A versatile labels program has been provided. See Reference Section 13.6. and HELP topic "labels".

TUTORIAL TWO

2.1 INTRODUCTION

In the previous tutorial we set up an Address Book file which you can use to store details of your friends and business contacts. In this tutorial we shall create a more complex application for processing the invoices and records of a small business.

If you worked through TUTORIAL ONE you will have had practise FORMATting a simple record layout, ENTERing data, creating a MEMO screen, accessing HELP screens and using simple methods of searching through your records provided by the SELECT option.

You will now be taken through further facilities provided by these options as well as Superbase options you have not as yet seen and will begin to see some of the power and flexibility which Superbase offers.

2.1.1 OBTAINING THE CUSTOMER RECORDS FILE

Place your Superbase disk in drive 0, and type

load "sb",8,1

and press RETURN. When prompted with

"Insert Data Disk and Press Return"

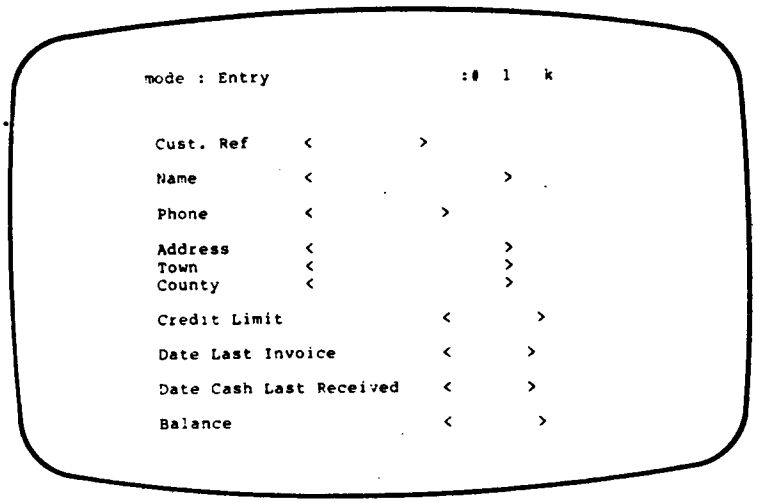
remove the Superbase disk, place your data disk in the drive unit and press RETURN.

When you are asked to specify the Database, choose 'Training' as before, but instead of selecting the 'Addresses' file select the file called 'Cust.Rec'. This is one of the two files we shall be using - one for the customers' accounts called 'Cust.Rec', and one for the invoices associated with these accounts, called 'Cust.Inv'.

This file is provided already formatted on the disk and so instead of being sent to the FORMAT option as last time, you will find yourself at MENU 1.

2.1.2 ENTERING INFORMATION INTO THE CUSTOMER RECORDS FILE

Choose the ENTER option and you will see that the layout of the screen for this file looks like this:



Enter information into the file from the list below, taking care that you enter the correct information into its corresponding field.

After each record has been filled out, press the RETURN key to file it as instructed at the top of the screen and then press the space bar to obtain another blank record.

FIELD NAME	FIELD CONTENT
Customer reference	BLO/10075 (the record key)
Name	Bloggs, Joe
Phone	01-765 9878
Address	12 Park Road
Town	St John's Wood
County	London NW1
Credit limit	100
Date last Invoice	14jun83
Date cash last received	01may83
Balance	(leave empty)

SHIFT for f2 f4 f6 f8

FIELD NAME	FIELD CONTENT
Customer reference	JON/10098
Name	Jones,Bill
Phone	01-765 3434
Address	67 Holland Ave.
Town	Kensington
County	London W8
Credit limit	250
Date last Invoice	01jun83
Date cash last received	09jun83
Balance	(leave empty)

FIELD NAME	FIELD CONTENT
Customer reference	SMI/10086
Name	Smith,Terrence
Phone	020-564 341
Address	89 Coventry Rd.
Town	Birmingham
County	West Midlands
Credit limit	250
Date last Invoice	07feb83
Date cash last received	01may83
Balance	(leave empty)

FIELD NAME	FIELD CONTENT
Customer reference	SMI/10090
Name	Smith,John,Gilbert
Phone	01-267 1156
Address	52 North Villas
Town	Camden
County	London NW1
Credit limit	50
Date last Invoice	23mar83
Date cash last received	01may83
Balance	(leave empty)

When you have entered this information into your records press RETURN instead of the space bar, and you will be returned to the MAIN MENU.

2.2 USING THE 'FILE' OPTION

We shall now take a look at the other file used in this application, the CUSTOMER INVOICES FILE.

From Menu 1 press RETURN to obtain the Menu 2. Select the FILE option by pressing the 'f1' key.

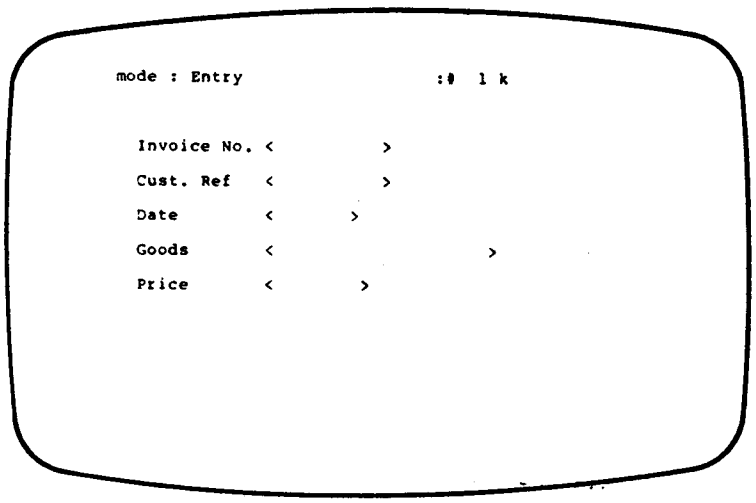
You will see the DATABASE CATALOG containing a list of the files in the current Database along with the number of records in each.

You have seen this screen before - it is the same screen as you obtain when starting up and are asked ' Enter Filename' It is also provided whenever you choose the FILE option to switch from one file to another within a database.

Type 'Cust.Inv' and press RETURN to obtain the CUSTOMER INVOICES FILE. You will be returned to Menu 1 but will notice that the FILE INDICATOR above the list of options now displays the name of the file we have just selected.

2.3 EDITING A SCREEN LAYOUT

Use the ENTER option again to have a look at the screen layout of the file. This is what you should see:



Before you enter any information into this file we are going to change the screen format by adding some fields of a type you have not yet encountered.

Screen layouts (sometimes called 'forms' or 'formats') may be changed at any time after the original set up, even if valuable data has been entered into the file.

When you have finished looking at the Invoice screen you can quit the ENTER option without adding data to the file by pressing the 'f1' key and then the 'Q' key. This will return you to Menu 1.

2.3.1 THE INVOICING APPLICATION

The application we are about to set up is for keeping track of invoices sent to your customers and the amount each of them owes you at any given time.

Each customer will have a record in the CUSTOMER RECORDS FILE which contains details about the customer including credit limit, the last date an invoice was sent, the last date you received cash, and the current balance of the account.

Every time an invoice is sent to a customer, this will be stored in the CUSTOMER INVOICE FILE. By using the two files together we shall be able to see exactly who owes you money and how much. The speed and sophistication of this type of operation is an important benefit of the Superbase system.

You will notice that at this stage the layout of the Invoice records is very rudimentary. There is as yet no means of taking account of either tax or any discount your customer may be due.

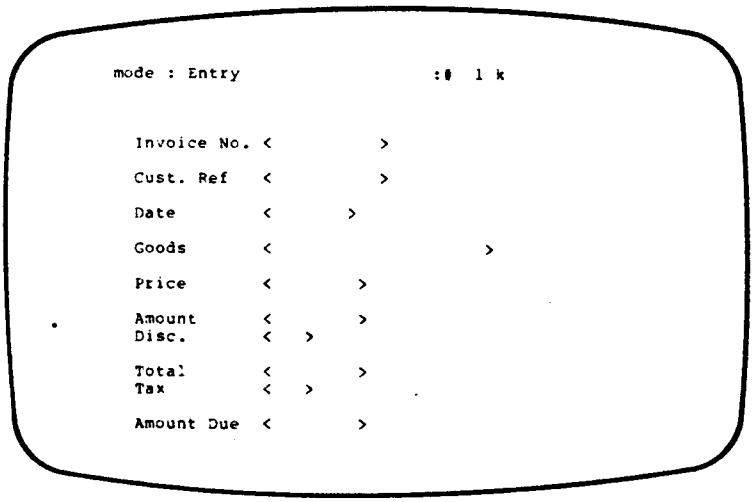
In the following section we shall make use of the FORMAT option to add these features to the Invoice records in such a way that Superbase makes the necessary calculations automatically for you.

2.3.2 SETTING THE RESULT FIELDS AND THE CONSTANT FIELD

From Menu 1 press RETURN to obtain Menu 2 and then select the FORMAT option by holding down the SHIFT key and pressing the top function key to get 'f2'.

You will see the Customer Invoice screen layout with the words 'Mode: Format' at the top of the screen.

Below is a diagram of the screen layout as it will be after we have made our modifications:



The first thing for you to do is to type the extra FIELD NAMES onto the screen. Use the diagram to get them all in the right place.

RESULT AND NUMERIC FIELDS

When you have done this, place the cursor a little along from the word 'Amount' and press the 'f1' key followed by 'R' for 'Result.' You should see the words 'Set Result' at the top of the screen indicating that this is to be a RESULT FIELD.

A Result Field is a field that will contain a number calculated by Superbase from information in the other fields in the record. In a moment we shall see how to specify the calculation we want.

Move the cursor along four spaces to the right and type a decimal point, then move the cursor two spaces right and press RETURN. The message 'End Set' will be displayed at the top of the screen.

The next field to be set is a numeric field. Numeric fields may only store the characters 0 to 9, the decimal point, and the '+' or '-' sign (only the '-' sign is displayed).

Move down a line and set the 'Disc.' field as a numeric field by pressing 'f1' followed by 'N'. The first numeric position is displayed. Move the cursor to the right two places and you will see # signs appear. As with Result fields, the cursor-left arrow may be used to delete # signs. Make the 'Disc.' field three characters long and press RETURN.

Now skip a line and create the 'Total' field exactly as you did the 'Amount' field.

A CONSTANT FIELD

Next move down to just along from the word 'Tax', press 'f1' followed by 'C' and the message 'Set Constant' will be displayed.

A CONSTANT FIELD is a field which contains the same value throughout the file. Using such a field saves you from typing information such as the tax rate over and over again for every record.

Move the cursor along right for three spaces and then obtain the 'End Set' message by pressing RETURN.

We need one more Result Field for the amount due. Set this field as you did for the 'Amount' field above, with four figures before, and two after, the decimal point.

When you have done this, press the 'f1' key followed by the 'RUN/STOP' key to signify to Superbase that you have finished editing the screen layout.

2.3.3 ENTERING THE CALCULATIONS

You should now see Superbase replacing the FIELD MARKERS one by one with angle brackets.

THE 'AMOUNT' CALCULATION

When it comes to the 'Amount' field it will pause and the message 'Enter Calculation' will be displayed. We want the 'Amount' field to contain the price of the individual item multiplied by the number of items ordered, so type now '[Goods]*[Price]' and press RETURN.

The square brackets indicate that it is the contents of the fields named inside the brackets that we want multiplied together and the asterisk (*) is simply the multiplication sign.

Note that one of the fields specified in the calculation, the field named 'Goods', is in fact a TEXT FIELD. This field will contain information such as '2 Calculators, model HY100' which includes both numbers and text. For the purposes of the calculation Superbase will pick out the first number in the field (in this case '2') and treat that as the numeric value of the field.

THE 'TOTAL' CALCULATION

You will next be asked to enter the calculation for the 'Total' field. The value we want here is obtained by subtracting the discounted portion of the total from the original 'Amount'.

One way of doing this calculation is therefore '[Amount]-[Amount] * [Disc.]/100'.

We use square brackets as before to signify the contents of fields. Parentheses may be used to give priority to any part of a calculation.

The calculation will perform these steps:

1 Divides the content of the 'Disc.' field by 100 to convert it to a fraction.
2 Multiplies that by the contents of the 'Amount' field to obtain the discount due.
3 Subtracts this from the contents of the 'Amount' field to arrive at a final amount due.

Enter this calculation and press RETURN.

THE 'TAX' CONSTANT FIELD VALUE

The next calculation you will be asked to enter is for the 'Tax' field. Since we set that up as a CONSTANT FIELD you need to enter the tax rate that you want displayed in all of the records.

Type in '15%'. Constant Fields are text fields so for the purposes of calculations Superbase will ignore the '%' sign. We shall therefore have to convert the '15' into a percentage when we use this field in subsequent calculations.

THE 'AMOUNT DUE' CALCULATION

Finally you will be asked to enter the calculation for the field which is to contain the total amount due after discount and tax.

The calculation is '[Total]+[Total] * [Tax]/100. This calculation makes use of another Result Field, the 'Total' field specified just now. The 'Amount Due' calculation performs the following steps:

1 Divide the content of the 'Tax' field (a CONSTANT FIELD which is a type of text field) by 100 to convert it to a fraction.
2 Multiply that by the content of the 'Total' field to obtain the amount of tax due.
3 Add that onto the 'Total' field to give us the amount due including tax.

Type in this calculation and press RETURN. This concludes the process of specifying the calculations for Calculated Result fields.

After a short pause you will be asked: 'Allow Duplicate Keys?' This is explained next.

2.3.4 UNIQUE KEYS AND DUPLICATE KEYS

A Superbase file may have either DUPLICATE or UNIQUE keys. The KEY to a record is, as you will remember, the field that is used for fast access to a record in a file. Usually it is best to keep every key in a given file UNIQUE: that is, not matching any other key in the file. Partially matching keys are fine, and may come in very useful in some situations. And it is also all right to give the same key to records in different files.

DUPLICATE keys are keys within the same file that are exactly the same. Suppose you want to use peoples' names as keys - as in the Address Book example in Tutorial 1. If you know two 'Smiths', you can only give the records for the two individuals the same key if you have previously replied 'y' to the 'Allow Duplicate Keys?' question at the end of the FORMAT Option.

The major drawback of duplicate keys is that when retrieving records for display or printing it can be difficult to access the second record or subsequent records with the same key as another. There are solutions to the problem, and you do not need to consider it further at this stage.

Remember that you can use the FORMAT Option and leave a file unchanged except for changing the reply to this question at the end of the process.

In the present application, both the Customer Records file and the Customer Invoices file will have UNIQUE keys.

The message at the top of the screen is asking if you want this file to have duplicate keys. Press the 'N' key for 'No'. The field names and markers will be erased, leaving descriptive text (including extra field descriptions) visible. When processing is complete, you will be returned to Menu 1.

2.4 ENTERING INFORMATION INTO THE CUSTOMER INVOICE FILE

From Menu 1, press the 'f1' key to obtain the ENTER Option.

Now enter the information below into the CUSTOMER INVOICE file. Note that the Result fields we set in the previous section do not appear on the list. This is because the contents of the fields are calculated and entered automatically by Superbase.

Also notice that the entry for the 'Tax' field does not appear. This is because it is calculated automatically from the value entered in the CONSTANT field during the FORMAT Option. Superbase will enter this value into every record, unless you override it yourself during the ENTER Option. Leave the 'Disc.' field blank.

Remember that after entering each record you will be told: 'Press Return to Store'. Do so, and then press the SPACE BAR to enter the next record. When you have finished, press RETURN instead of the space bar, and you will be returned to Menu 1.

FIELD NAME	FIELD CONTENT
Invoice No.	01026
Cust.Ref.	BLO/10075
Date	14jun83
Goods	2 Calculators
Price	28

FIELD NAME	FIELD CONTENT
Invoice No.	01009
Cust.Ref.	JON/10098
Date	01jun83
Goods	1 Radio
Price	12

FIELD NAME	FIELD CONTENT
Invoice No.	00627
Cust.Ref.	SMI/10086
Date	07feb83
Goods	1 Portable TV
Price	180

FIELD NAME	FIELD CONTENT
Invoice No.	00907
Cust.Ref.	SMI/10090
Date	23mar83
Goods	12 Tapes
Price	2

2.5 MULTIPLE SCREENS

It would be useful if our Invoice Records had fields to indicate whether or not the Invoice has been paid, how much has been paid, and when the payment was made. We could also include a field to tell us if the payment has been added to the BALANCE field in the Customer Record file.

The only apparent difficulty is that the record screen has no more room for extra fields. In fact, this is not a problem since records can include up to four separate screens. So far we have been working with single screen records, but we shall now add an extra screen to the Invoice file.

REDEFINING THE FILE FORMAT

To do this we will have to edit the screen format again, so press RETURN to get Menu 2 and then hold down the SHIFT key and press 'f2' to select the FORMAT option.

OBTAINING EXTRA SCREENS

Once you have the Customer Invoice Format on the screen, press 'f1' and then the '+' key. The message 'Forward Screen' will indicate that you have moved forward to a new screen. This is how you obtain an extra page for your records.

The '+' command can also be used in the SELECT option to obtain the next screen. The converse command '-' is used for returning to the previous screen.

SETTING FURTHER FIELDS

TEXT

Using the screen diagram below, type the field name 'Paid', move the cursor along a couple of spaces and set a TEXT FIELD by pressing 'f1' followed by 'T'. Make the field three characters long to allow you to enter 'yes' or 'no', and then set the end of the field by pressing RETURN.

NUMERIC

Next type the name for the 'Total-paid' field, move the cursor along and set a NUMERIC FIELD by pressing 'f1' followed by 'N'. Move the cursor along four spaces, type a decimal point (period), and move the cursor another two spaces to allow for two figures after the decimal point. Now press RETURN to set the end of the field. Then type the name for the 'Outstanding' field and set a RESULT field by pressing 'f1' then 'R'. Set this field as the last one to give four figures before, and two after the decimal point.

Now type the name for the 'Last-paid' field and set another NUMERIC FIELD like the 'Total-paid' field above.

DATE

Follow this with the field name 'Date-paid' and set a DATE FIELD using 'f1' followed by 'D'.

TEXT

Finally type in the 'Pending update?' field name and set a TEXT FIELD just like that for the 'Paid' field above.

UNIQUE FIELD NAMES

The reason for the hyphen in the 'Total-paid', 'Last-paid' and 'Date-paid' field names is that Superbase takes the last word before the field as the field name, taking spaces as signifying the end of one word and the beginning of another. If we had left out the hyphens, then all three of these fields would have been called 'paid'.

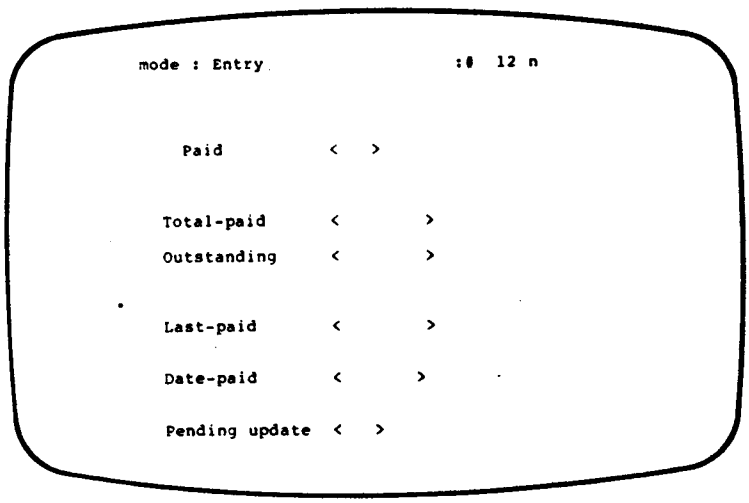
> **This is an important Superbase rule: each field in a record must have a field name. If this is not unique, then the contents of the original field will be copied into all subsequent replicas.**

ENTERING THE CALCULATION

Check the appearance of the new screen. When you are satisfied, press the 'f1' key followed by the 'RUN/STOP' key. All previously entered calculations will be displayed in order. Press RETURN to accept them. You will then be asked to enter the calculation for the 'Outstanding' field which was set as a RESULT FIELD.

Type '[Due]-[Total-paid]' and press RETURN. Answer 'N' for 'No' when asked 'Allow Duplicate Keys?'

The second screen of the record should look like this:



2.5.1 ADDING INFORMATION TO THE SECOND SCREEN

We now need to enter information into the second screen of our Customer Invoice Records. Since we want to add this information to already existing records rather than produce new records as with the ENTER Option, we instead use the REPLACE command as we did in section 1.7.4 above.

From Menu 1 press the 'f2' key to obtain the SELECT Option.

You will see the Select Menu on your screen. Press 'F' to select the first record and then press 'R' for 'REPLACE'. The cursor will be in the second field. Using RETURN or the cursor-down key, move down through the fields.

You will see that when you leave the last field on the screen, the next screen will be displayed with the cursor in the 'Paid' field at the top.

Type information into the fields from the list below, pressing RETURN to file each record and then 'N' for NEXT to obtain the next record to be replaced. Press 'R' each time to replace the record in the file. At the end press RETURN twice to return to Menu 1.

INVOICE NO. 00627

FIELD NAME	FIELD CONTENT
Paid	No
Total-paid	LEAVE BLANK
Last-paid	35
Date-paid	01may83
Pending Update	Yes

INVOICE NO. 00907

FIELD NAME	FIELD CONTENT
Paid	NO
Total-paid	LEAVE BLANK
Last-paid	10
Date-paid	01may83
Pending Update	Yes

INVOICE NO. 01009

FIELD NAME	FIELD CONTENT
Paid	Yes
Total-paid	LEAVE BLANK
Last-paid	12
Date-paid	09jun83
Pending Update	Yes

INVOICE NO. 01026

FIELD NAME	FIELD CONTENT
Paid	No
Total-paid	LEAVE BLANK
Last-paid	0
Date-paid	LEAVE BLANK
Pending Update	Yes

2.6 SEARCHING THROUGH THE INVOICE FILE

INDIVIDUAL INVOICES

Since the key field of the Customer Invoice file is the invoice number, the easiest way to find a particular invoice is to do a KEY SEARCH using this number, similar to the key search by name that we did on the Address Book file in Tutorial One.

This does of course assume that you know the invoice number of any invoice you want to find. But most inquiries for a single invoice will arise from a query about the invoice itself, so either you or your customer will most probably be looking at it and able to read the invoice number straight off the document. Try this KEY SEARCH method now if you wish, then return to Menu 1 and carry on with the next step.

SEARCHING FOR RANGES OF INVOICES

Let us suppose that you in fact want to see all invoices which have had payments made on them, but which still require the amount last paid to be added onto the balance in the Customer Records file. From Menu 1 press the 'f2' key to obtain the SELECT MENU and then press 'M' to obtain the MATCH option.

Just as in Tutorial One, you will now see the Customer Invoice record displayed with the cursor in the first field. Move the cursor to the 'Last-paid' field on the second screen and type '>0'. This will only select records with an amount 'greater than' zero paid on the invoice.

Now move to the 'Pending Update' field. The match you require '=yes' is too long for the field so type <- (back arrow). At the end of the MATCH you will be prompted with 'Pending Update' so that you can enter '=yes'. This will select only those invoices which have had the word 'yes' placed in this field to indicate that the amount is to be added to the 'Balance' field of the Customer Records file.

Now press SHIFT RETURN and Superbase will find all of your invoices which match BOTH of the criteria you specified.

When the first such record is displayed, verify that it matches the specified criteria by using the '+' key to see the second screen. When you are satisfied press the 'M' key to find the next record which matches, continuing to do so to the end of the file. Because you used the '+' key, the display will now always start at the second screen unless you use the '-' key to revert to the first screen. Press RETURN twice to obtain Menu 1.

2.7 FINDING A KEY LIST

You have just inspected all the invoices that have not yet been used to update the Customer Records file. To have a list of them would make the updating much easier.

This is where the FIND Option comes in. The option can be selected from

Menu 1 and is operated just like the MATCH option as far as specifying the match criteria is concerned.

However, instead of displaying the matching records on screen, the FIND option creates a list of them and stores it. Since these lists are in fact just lists of the KEYS (i.e. contents of the KEY FIELD) of the matching records, we call them KEY LISTS.

FINDING THE LIST

From Menu 1, press the 'f3' key to obtain the FIND option and enter the same criteria as you did in the section above. Superbase will search through the file and make a list of all the relevant keys. When the list is complete, you will be returned to Menu 1.

NAMING THE LIST

Unless you actually give the list a name, as we shall see how to do soon, Superbase will give it the default name "hlist". The purpose of the 'h' is to make the list accessible from the HELP option. You may remember from the last tutorial that a Help Screen is just a memo screen with an 'h' prefixed to its name. Having our default list called 'hlist' gives us an easy way of viewing the list of keys itself simply by selecting HELP and typing 'list' when asked 'Which help required?'.

Try this now. You should see the list of invoice numbers corresponding to the records you have selected.

In the section after next we shall see how we can use the list to produce a display of the records, on screen or printer, whose keys appear on the key list.

Do not give your list the same name as a file in the database (such as "cust.rec), or you will overwrite the file format definition. To recover from this error, reformat the file using 'Format'.

2.8 OUTPUTTING RECORDS

THE OUTPUT PROMPT

Before you begin, type in the word 'down' on the Command Line and press RETURN. The reason for doing this will be explained later. Then, from Menu 1 press the 'f4' key to obtain the OUTPUT Option. This prompt will appear on the command line:

```
Enter:  all/from "list" (item list....)
```

The OUTPUT Option is for displaying or printing information from your files. You can choose to output from either all the records in the file, or from only the records whose keys appear in a list you have made with FIND. This is the meaning of the 'all/from' part of the prompt. The name of the list is then given inside quotation marks, or these may be left empty (i.e. "") if you wish to use the default list "hlist." Then you specify the fields from the record and other items to be output. There are many ways of outputting, and what is not explained here is discussed in the reference section.

TUTORIAL - LEVEL TWO

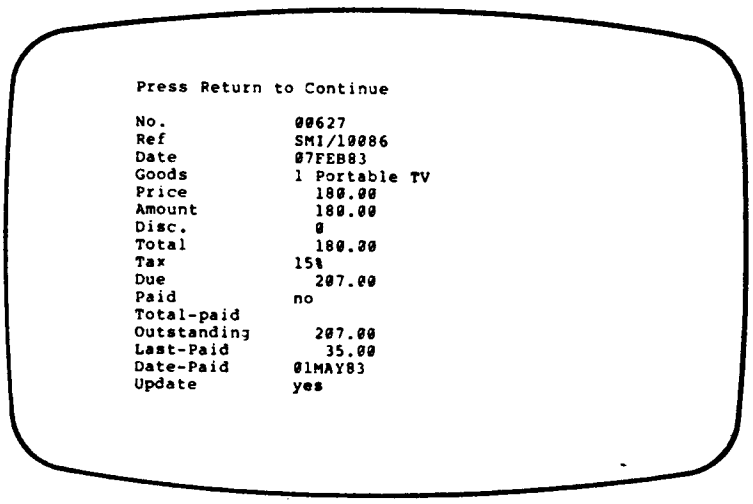
'ACROSS' AND 'DOWN'

The direction of the output display or print is determined by the ACROSS and DOWN commands. Either of these may be typed in on the command line by itself and will then set the direction of all output until the converse command is received. The default direction on start-up is ACROSS, which is why you were asked to type 'down' in the previous paragraph.

OUTPUT WITH NO KEY LIST - ALL FIELDS

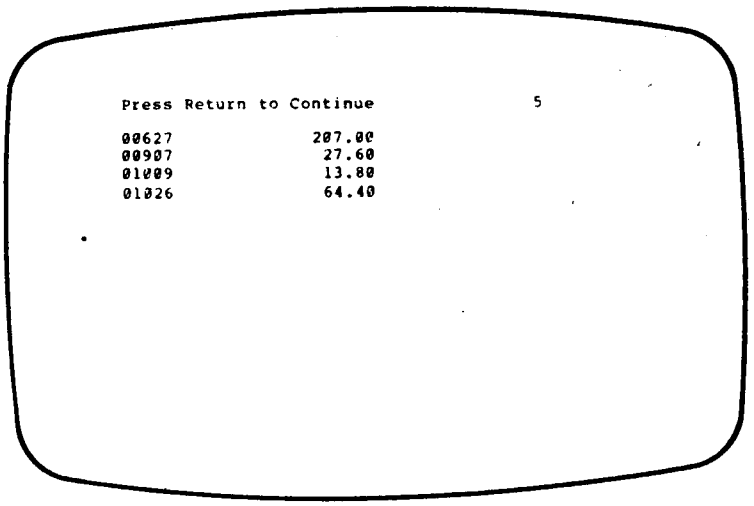
Now try out a display of all fields from all records. Type 'all the records' and press RETURN. You should see the information from your first record displayed on the screen as in the diagram below.

Press RETURN and you will see the information from the next record displayed.



OUTPUT WITH NO KEY LIST - SELECTED FIELDS ONLY

You may, however, wish to see only selected fields. Try selecting the OUTPUT Option again and typing 'across all the records [no.][due]' followed by RETURN. The square brackets contain the names of the fields to be displayed and we are asking again for ALL of the records. Note that this time as many records as will fit are displayed on the same screen as in the diagram below.



OUTPUT FORMATTING COMMANDS

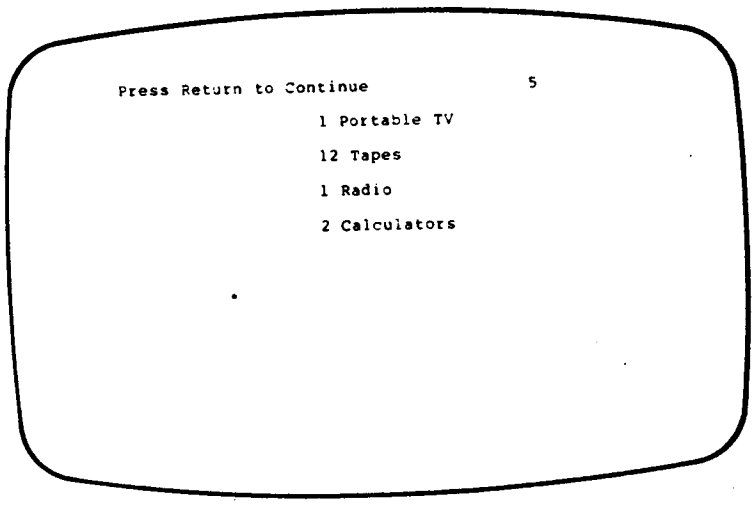
If we wish we may improve the format of the display by using the following command characters:

'@x'	Displays next item at column x.
'@x,y'	Displays next item at column x and row y.
'&'	Cuts off all trailing spaces.
'&x'	Truncates contents of next text item to x characters.
'&x,y'	For numeric items. It formats the numbers to x figures before and y characters after the decimal point.

OUTPUT ALL FIELDS AT COLUMN 20

Let's try out some of these command characters now. Select the OUTPUT Option again and type 'all the records @20 [goods]' and press RETURN.

This is what you should see:



OUTPUT ALL FIELDS AT COLUMN 20, ROW 15

Now try 'all records @20,15[Goods]'. Can you see why only one record is displayed at a time in this case? It is because although information from each record can be displayed in the same column on the same screen (one below the other as in the previous example), it is impossible for information from each record to be displayed in the same row AND the same column on the same screen without each item overwriting the item before it.

OUTPUT USING TRUNCATION AND DECIMAL POINT FORMATTING

Experiment with the truncator command ('&'). Try out

'all the records &6 [No.] &4,1 [price]' and

'all records &[Goods][Ref]'

DISPLAY OVERFLOW

Note that the numeric truncator (&x,y) must specify a value for 'x' at least as big as the number of figures BEFORE the decimal point in the field to be displayed.

If 'x' is too small, then a row of '#' symbols will be displayed instead to show you that the numbers overflow the format you have specified for them.

NUMERIC ROUNDING

If the value of 'y' which governs the number of figures displayed AFTER the decimal point is smaller than in the number to be displayed, then the number will be rounded up. For example, '&4,1 [price]' where the price is 82.69 would display '82.7'.

2.8.1 OUTPUT FROM A KEY LIST

We shall now combine the concepts learned in the previous two sections to produce a display of information from just those records whose keys we have stored in a KEY LIST.

Obtain the OUTPUT Option again and in response to the prompt type:

'the records from "hlist" [no.] [price] [outstanding]'

and press RETURN.

This is where the usefulness of the Key List facility starts to become apparent. Whereas the MATCH command allowed you to SELECT records which satisfy certain criteria, with the help of a Key List you can USE the selected records in various ways - printing them, displaying them, or processing them in other ways.

OUTPUT TO THE PRINTER

Although up to now we have been displaying our output on the screen, it is just as easy to have a hard copy printed out.

Make sure that your printer is connected and switched on. Obtain the OUTPUT Option again and type 'print from "hlist" [no.][goods][due]' and press RETURN.

Your printer should spring into action and provide you with printed output. Note that PRINT switches the direction of output from screen to printer, so that all output from now on will be to the printer until you specify otherwise.

Check this out by entering another output command such as 'all the [goods]'. To change back to a screen display we use the word DISPLAY as in:

'display records from "hlist" [no.] [price]'

Whichever of the commands PRINT or DISPLAY was used last will continue to be operative until the other one is used, either in the Command Line or in a Program (these are explained below).

THE DEFAULT KEY LIST

We mentioned earlier that 'hlist' is the name that Superbase gives to a Key List if no other name is specified. This is a useful feature that frees you from the need to think up and type in a new name every time you select a list. Its only disadvantage is that each time you create another Key List, that list becomes the current 'hlist' and the previous Key List is lost.

However, there is a simple and obvious way of retaining useful key lists - renaming them. Once you have given a name other than "hlist" to a Key List Superbase stores it on disk under that name for future use. All lists are automatically stored on disk, including the 'hlist', so the results of any selection are not lost when the system is switched off. You will be shown how to do this in the section after next, but first you need to know about DIRECT COMMANDS.

2.8.2 DIRECT COMMANDS

In Tutorial One we mentioned that Superbase has a dual control structure, that it is both Menu-driven and Command-driven. If you have worked through this tutorial you will now have had experience of both.

The expressions you entered as a response to the prompt given by the OUTPUT Option were your first experience of what we call COMMAND LINES.

All of Superbase's options and facilities can in fact be accessed without the use of the MENUS at all, just by typing Command Lines onto the message area at the top of the screen from either Menu 1 or Menu 2.

OUTPUT USING THE COMMAND LINE

From either Menu, type the Command Line 'Output all the records' and press RETURN. This has exactly the same effect as pressing 'f4' to obtain the Output option and then responding 'all the records' to the prompt.

A Command line always begins with a Primary Command such as those which appear on Menu 1 and Menu 2, and is followed by the Secondary Commands and expressions associated with the option corresponding to the Primary Command.

ABBREVIATING THE COMMAND LINE

The words 'the' and 'records' and the spaces are not actually needed for Superbase to understand your Command Line but serve simply to make the syntax more natural. You could have entered the last Command Line simply as 'Outputall' if you had wanted to.

You could even abbreviate further to the first letter of the command with the second letter SHIFTed as in 'oUaL' or 'oUfR"hlist"[goods][price]'.

The rule is that any command word can be reduced to the shortest possible unique form of the command, with a minimum length of 2 characters.

You may want to try to obtain the results you achieved earlier via the Menus by using command lines instead. Try 'file "Cust.rec"' then 'select first'.

COMMAND LINES WITH MORE THAN 1 COMMAND

A Command Line may include more than one Primary Command if they are both in different CLAUSES separated by a colon. For instance, the last two Command lines could have been two clauses of the same Command line, as in 'file "Cust.rec":select first'.

2.8.3 NAMING AND STORING A KEY LIST

To name a Key List so that it can be stored and used independently of the 'hlist' we have to issue the direct command 'find "listname" ' where 'listname' (which must be in double quotation marks) is any name you like up to 16 characters.

If you were to issue the command 'find "listname"' as above, you would find yourself presented with a MATCH SCREEN so that you can specify the criteria which governs membership of the list.

There is a way of creating the list completely from a Command Line however. The command is 'find "listname" where [field] is "<match criterion>";[field] is "<match criterion>

What goes into '<match criterion>' here can be '=Jones' or '>100' or any of the other Match Criteria available (for match criteria other than those you have met see Reference Section 6.2.7).

First use the 'FILE' command to select "Cust.inv", and then try the following command to create a permanent list like the 'hlist' we have been using in previous sections:

'find "update-list" where [last-paid] is ">0";[update] is "=yes"'

Once a Key List has been created with a name in this way it can be used any time in a Command line such as 'output from "update-list" [goods] [disc.]' or whatever you wish. Try out this command yourself.

2.9 REPEATING COMMANDS WITH THE '←' KEY AND STORING THEM

Just as there are two ways for Superbase to store a Key List: temporarily under the name of 'hlist', or permanently under some other name, so too a command line can be held either in a short-term or a permanent fashion.

In order to recall the last Command Line entered we use the left-arrow key at the top left of the keyboard. The Command Line will be displayed at the top of the screen, either to be executed again (press RETURN) or to be modified (type modifications).

To prepare, type 'f2' followed by 'f' and then press RETURN twice to position to the start of the file and return to Menu 1.

When you have done this, try typing in the following Command Line followed by RETURN:

```
'select next:print[no.];"amount outstanding is";[outstanding]'
```

Note that between the two fields to be printed we have a string of words between double quotes. This is called a TEXT STRING and such Text Strings may be inserted before or after any field to be printed or displayed.

You will have found that the last Command Line caused the Customer reference number and the amount outstanding on that invoice to be printed out from the second record on the file. But what if we wanted to continue from the next record?

Press the left-arrow key and you will see the command displayed again at the top of the screen. Press RETURN and the command will be re-executed. Because of the 'Next' command, information from the next record will be printed.

Press the left-arrow key followed by RETURN twice more and watch the results.

The reason for the left-arrow displaying the Command Line instead of immediately executing it is not only so that you can modify it but also so that you can store it permanently. How this is done will be explained in the next section.

2.9.1 LINE NUMBERS AND PROGRAMS

You have seen how the current Command Line can be recalled by means of the left-arrow key to be used again. However, once a new Command Line is typed in, the old one is lost.

To store a Command Line permanently for future use we must first give it a LINE NUMBER. Try it out now by pressing the left arrow key to obtain your current Command Line and then use the INSERT key (SHIFT+INST/DEL) to make room before the Command and type the number '100' followed by RETURN.

Type 'select first' and press RETURN to position to the start of the file again and then press 'f7' or type 'execute' and press RETURN. You will see the program run extremely fast.

By storing a Command Line with a Line Number in this way, you have created a simple PROGRAM.

You can execute the Program at any time from Menu 1, either by selecting the EXECUTE option ('f7') or by typing the command 'Execute' and pressing RETURN.

As is detailed in the Programming Section, a Program may consist of a number of Command Lines which will be executed one after the other.

As it stands at the moment, the Program we have just created does its work on only one record from the file at a time. This is why we had to continually press the left-arrow key to repeat the command for each of the records.

To make the Program operate on all of the records we need to modify it to:

```
100 output print all the records [no.];"amount outstanding
is";[outstanding]'
```

Let us do this now as an illustration of how a Program can be modified.

From Menu 2, press 'f5' to obtain the PROG option. You will see your Program displayed on the screen with the cursor in the HOME position. Type the modifications as above and then press 'f1' followed by RUN/STOP.

To see that the Program file is now as modified, select the EXECUTE option from Menu 1 ('f7') and you will obtain a print-out of Invoice Numbers and the amount outstanding on each of them for all of the invoices in the file.

To stop a program you must use the STOP key.

Further details of how to create Programs, including a Program to update the balance field of the Customer Records file with information from the Customer Invoices file, are given in the first part of the Programming Section.

You are strongly advised to BACK-UP your data disk regularly. A particularly important time to do BACK-UP is before any major system update to ensure the integrity of your data. BACK-UP is done from the MAINTAIN Option. See Reference Section 15.4.

TUTORIAL THREE

3.1 INTRODUCTION

In the previous tutorial we looked at using the FILE option, setting RESULT FIELDS and CONSTANT FIELDS, using MULTIPLE SCREENS, various SEARCH techniques, creating a KEY LIST and OUTPUTTING information from records.

The Invoicing application we set up can be made more complex in a number of ways, and in the section on PROGRAMMING Superbase a program is provided that will update the BALANCE field in the Customer Records file with information from the Customer Invoice file.

In the tutorial which follows, however, we shall not be discussing the techniques of creating programs. When you are ready, you can study the more advanced aspects of Superbase at your leisure. For now we shall concentrate on four more of Superbase's options: the CALC, BATCH, SORT and REPORT Options.

3.2 PERFORMING CALCULATIONS ON THE FIELDS

If you have just loaded Superbase, select the 'Training' database and from the Database Catalog that appears on the screen load the Customer Invoice file, 'Cust.Inv'.

Using the ENTER Option as detailed in Tutorials One and Two enter a number of records of your own devising, making sure that each customer has more than one invoice.

When you have done this, use the RETURN key to obtain Menu 1 and press 'f5' to enter the CALC Option.

The CALC Option is for carrying out calculations of any kind, using any of the functions provided by BASIC in addition to the normal arithmetic operators. CALC can be used as a simple calculator. Type 57*17 and press RETURN. The answer will be displayed on the Command area.

One advantage of the CALC facility is the ability to use FIELDS from the current record in these calculations, either to modify their values and STORE them as modified, or to experiment with 'What If' conjectures by seeing the effect of changing the value of one field on the values of other fields in the record.

For example, type the following calculation and press RETURN:

```
[disc.]=15;[Price]=[Price]+[Price]*0.1
```

The CALC line changes the discount to 15% and increases the price by 10%. Verify that the [Due] and [Outstanding] fields in the current record, which are result fields, have indeed changed by pressing 'f2' and then 'C' from Menu 1. Press RETURN twice to return to the menu.

The reason we cannot display these fields in the CALC Command line itself is that field references within a single command are always to the original value of the field. Only by typing a full Command line with colons to separate the individual commands to modify and then to display a field (or its derived result fields) can you achieve both modification and display in one operation.

This calculation will have no permanent effect on the record unless you next give the command 'STORE' from the Menu to store the modified record on disk.

As another example, let us suppose that you want to find out how much one of your customers would need to pay you per month to clear the amount outstanding on a particular invoice in a year.

To set this up, use the SELECT option as detailed in Section 1.7.4 of Tutorial One to REPLACE a record so that you can set the price high enough and the total paid low enough for the 'Outstanding' field to be a considerable amount.

Select the CALC option again and type the following calculation:

```
x=[Outstanding]/12;&[Ref] "will have to pay" x "per month"
```

In this example we have used a BASIC VARIABLE to store the monthly amount and are displaying its value along with text to make clear what it is supposed to be.

Note that fields, text, or variables that are to be simply displayed must always be placed at the end of the CALC line.

3.3 PERFORMING BATCH OPERATIONS ON THE FILE

Useful though it is for 'What If' conjectures on the current record it would be tedious if we had to do a CALC followed by a STORE on every record if we actually wanted the modifications to remain in the file.

Fortunately, Superbase has another option for this purpose. The BATCH Option will carry out calculations on the fields just like the CALC Option but it will do so on ALL of the records in the file or on those records in a specified KEY LIST, and will automatically store the modifications permanently on disk.

Before we use the BATCH option we need to set up a Key List of all those records needing updating. Using the technique you learned in Tutorial Two Sections 7 and 8.3, type 'find "update-list"' from the Menu and then use the FIND Option specifying only 'yes' in the 'update' field.

From Menu 2 select the BATCH Option by pressing the 'f3' key. You will be prompted with 'all/from "list" (item list...)'.

Respond to the prompt with:

```
from "update-list" [total-paid]=[total-paid]+[last-paid];[update]="no"
```

Press RETURN and the updating will commence with the message 'Processing' displayed on the screen.

When you are returned to the Main Menu, check that the updating has indeed been carried out by using the SELECT option to browse through the records as outlined in Tutorial One, Section 1.7.1.

3.3.1 RUNNING TOTALS

Another use that the BATCH Option can be put to is that of displaying running totals through the current file.

We saw in the CALC section above that we can store the values of fields in BASIC VARIABLES which can then be displayed. To produce a running total what we do first is define a BASIC variable by giving it a name, say 't', and set it to an initial value of zero. We then look at all the records in turn, finding the required value and adding it to 't'. At the end the value of 't' is the total for all the records.

First type in the Command 'calc t=0' and press RETURN. This sets the variable 't' (which we invent simply by naming it) to a value of zero. Now enter the BATCH Option gain and type the following BATCH command.

```
all the records t=t+[outstanding]; "Total outstanding is ";t
```

The records will be processed one by one and the value of each OUTSTANDING field will be be added into 't' which will be displayed increasing in value until it reaches its final value at the end. If you now wish to see the final total type 'display t'.

In Section 5 of this tutorial we shall see how sophisticated REPORTS can be produced which also display totals across all of the records, but first we must take a look at the SORT Option.

3.4 SORTING RECORDS

As its name suggests, the SORT Option is for producing lists of the keys to records sorted into an order different from their normal alphabetic order. The records in the file remain unchanged and in their original order. The list of sorted keys is used in other operations such as OUTPUT and REPORT.

From Menu 2 press the 'f4' key (remembering to hold down the SHIFT key) to obtain the SORT Option.

You will be prompted with:

```
all/from "list" (item list....)
```

Type in the response:

    all on [ref];[due] to "sorted list"

and press RETURN.

The message 'Processing' will be displayed and after an interval you will be returned to the Main Menu.

Your response to the prompt signified that you wanted to produce a list of all the records sorted in order of the Customer Reference code ([ref]), and for those records with the same reference code, in order of the amount due ([due]).

The list you have named 'sorted list' will now contain the Keys of the records in this sorted order.

If you had included the letter 'D-' before the word 'on', the list of sorted keys would have been produced in DESCENDING order, i.e. beginning with the alphabetically last one.

To verify that the SORT has taken place use the OUTPUT Option to Display From "sorted list" using the instructions for this option you were given in Section 2.8.1 of Tutorial Two.

3.5 PRODUCING REPORTS

We shall now conclude this tutorial by using the REPORT Option to create a report on how much each customer owes and has paid on the goods bought.

For a more comprehensive account of the powerful REPORT facility see Reference Section 13.

From Menu 1 use the 'f6' key to obtain the REPORT Option, remembering to hold down the SHIFT key.

The REPORT GENERATOR is entirely prompt driven and constructs for you a Report Program which when executed will produce the report.

REPORT FILE

The first prompt is:

    Enter File to Report on

Type "Cust. Inv" (you must use double quotation marks) and press RETURN.

REPORT TITLE

You will then be asked:

    Enter Report Title

Enter "REPORT ON GOODS SOLD" and press RETURN. Note that the title must be within double quotation marks. This is the heading of your report and will appear at the top of each page of the report.

In case you wanted a title larger than would fit in the Command Area you are asked:

Any More?

Type 'N' for no.

TOTALS AND SUBTOTALS

The next prompt is:

Enter Total Calculation

These are the calculations required for Totals at the end of the report and subtotals during the report itself. They are entered in the same way As the total we specified in the BATCH option at 3.3.1 above. You have ten special Total Variables, t0, t1....,t9 and ten Subtotals Variables s0, s1....,s9 to use in this option. Type:

s0=s0+[total-paid];t0=t0+[total-paid]

and press RETURN. You will be asked 'Any More?' in case the line of totals and subtotals is longer than can be fitted in the two line command area. Reply 'y' as this is the case now. Then type:

s1=s1+[outstanding];t1=t1+[outstanding]

You will be asked 'Any More?' again. This time reply 'n'. We have specified that we want to total and subtotal the amount paid and the amounts outstanding on all of the invoices.

SUBTOTAL BREAK POINT

After specifying totals and subtotals you will be prompted with:

Enter Field for Subtotal Break

Type '[ref]' and press RETURN. The field for subtotal break determines when the subtotals are to be printed. Each time the field specified changes its value we will get a subtotal. By specifying the Customer Reference code we will have a different subtotal for each customer.

Next there is a prompt of:

Enter Subtotal Text

This is the text you want printed with the subtotals to indicate what it is. Type

@1,0"Total Paid:" @19 s0 @1 "Total Outstanding:" s1

and press RETURN.

You will be asked 'Any More?' in case you want more subtotals, but since we do not, type 'N'.

KEY ORDER OR USE A LIST

A prompt will next be given to find out whether you want the records from the file in the order in which they are stored (i.e. their key order - this would be ALL the records), or those from a Key list. Although we do want to use all of the records we do not want the records to be printed in key order. Instead we want to use the Key List 'sorted list', so that the information is grouped according to customer and amount (see the section on SORTING records above), so respond with:

```
from "sorted list"
```

and press RETURN.

REPORT DETAIL

The detail that will form the main substance of the report is prompted for next with:

```
Enter Report Detail
```

Type

```
@1,0 [ref] [goods] [due] [outstanding]
```

and press RETURN. Again you will be prompted 'Any more?' Reply as before. It is the contents of these fields we want to appear for each record used in the report.

END OF REPORT

The final prompt for the specifications of the report itself is:

```
Enter End of Report Text
```

This is the information you want displayed at the end of the report including the final totals. Type

```
@1,0 "Report Total Paid:"t0 @1 "Report Total Outstanding" t1
```

and press RETURN. The 'Any more?' prompt is presented for the last time.

STORE THE REPORT

The program created by the Report Generator to your specifications will now be displayed and you will be asked:

```
Save Report Definition
```

Type 'Y' for 'Yes' and the following prompt will ask you what you want the Report Program to be called on disk:

```
Enter Report Name
```

The screen shows

```
save "
```

Type 'Sales Report' (ending with double quotation marks), and press RETURN. The Report Program will be stored for you to use at any future date.

If you wish to print the report type 'print' first to switch output to the printer. To print the Report itself press the 'f7' key from Menu 1 or type in the command 'Execute' directly from either Menu.

There may be errors in your Report Program. This is reasonable, since you have just used the Superbase Report Generator for the first time. However, you will wish to correct them so that the Report will run smoothly. The Program should look like this:

```
100  report "cust.inv"
200  title "REPORT ON GOODS SOLD"
300  total s0=s0+[total-paid];t0=t0+[total-paid]plus
400  s1=s1+[outstanding];t1=t1+[outstanding]
500  subtotal [ref]@1,0"Total Paid:"@19s0@1"Total Outstanding" s1
600  detail from "sorted list"@1,0[ref] [goods] [due] [outstanding]
700  endreport @1,0"Report Total Paid:"t0@1"Report Total Outstanding" t1
```

Correct your Report Program so that it looks like this, using the PROG editing facilities. You must then re-save the Program from the Main Menu.

HINTS AND TIPS

You are strongly advised to BACK-UP your data disk regularly. A particularly important time to do BACK-UP is before any major system update to ensure the integrity of your data. BACK-UP is done from the MAINTAIN Option. See Reference Section 15.4.

On changing data disks always ensure that you reselect your database by typing the command 'database' on the command line and following the prompts.

Do not duplicate the name of a database. If you use the name of an existing list or file, it will be overwritten by the new list.

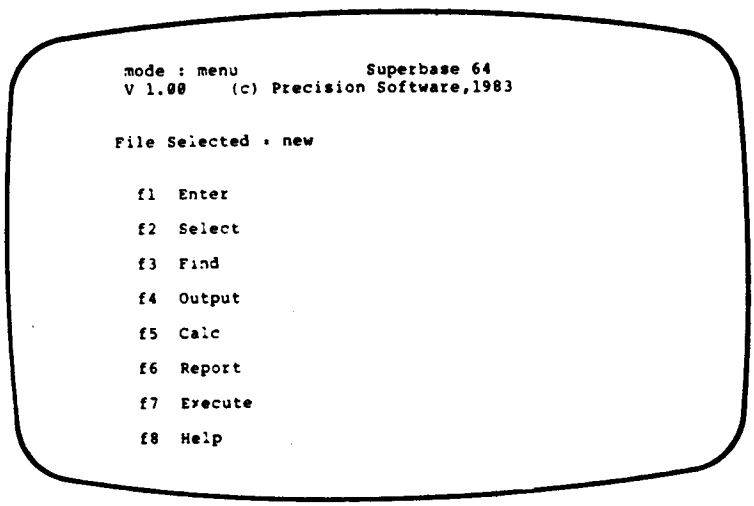
LABELS A versatile labels program has been provided. See Reference Section 13.6. and HELP topic "labels".

1 THE MAIN MENU - MENU 1

1.1 USING MENU 1

Superbase combines great sophistication with simplicity of operation. By means of the various Menus you can access all of the options provided in an easy and natural manner.

MENU 1 is your entry point into Superbase's most often used facilities, each of which is described briefly below beside the Function key used to obtain it.



1.2 THE OPTIONS

ENTER This is the option used to ENTER information into your files. Deselect with 'f1' followed by 'Q'. See Section 5.

SELECT The SELECT Option is used to find and display any record stored in the database. It has been given a Menu of its own from which you can choose a wide range of facilities. Deselect with RETURN. See Section 6.

FIND The FIND Option is used to find records matching a particular set of criteria. It stores a list of the keys of these records to be used by other options such as SORT and OUTPUT. Deselect with 'f1' followed by 'Q'. See Section 7.

OUTPUT The OUTPUT Option is used to display or print information from all records or a previously selected list of records. Text as well as the contents of fields can be output and BASIC variables and calculations included. Deselect with RETURN. See Section 8.

CALC The CALC or CALCULATE Option is used to evaluate and/or display any expression. The full range of BASIC functions including trigonometrical functions can be performed. Results can be stored in fields or BASIC variables or simply displayed. Deselect with RETURN. See Section 11.

REPORT This REPORT Option provides you with a full set of commands for producing printed REPORTS from the information in your files. Deselect with RETURN. See Section 13.

EXECUTE The EXECUTE Option allows you to run previously designed PROGRAMS which can automatically perform whole sequences of operations on your files. Whole jobs can be carried out at the touch of a button. Deselect with RETURN. See Section 14.

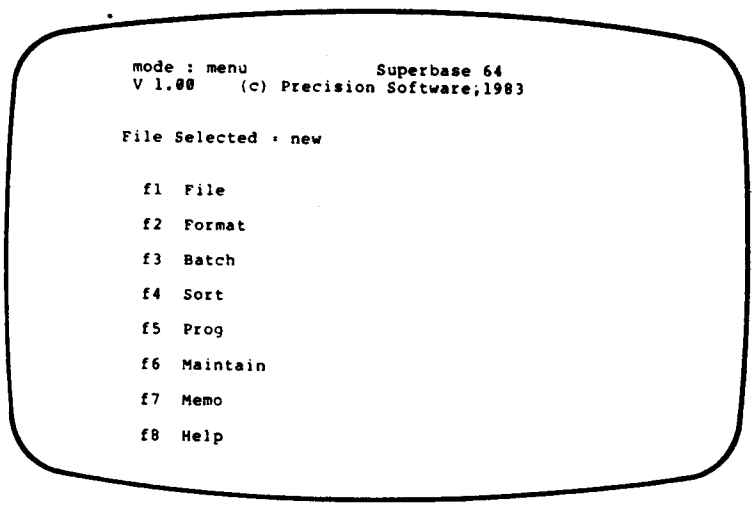
HELP The HELP Option is used to display a HELP SCREEN designed to give reference information and memory joggers for all of the major facilities offered by Superbase. Deselect with RETURN. See Section 17.

THE SECONDARY MENU - MENU 2

2.1 USING MENU 2

Menu 2 provides a range of further options, mainly those capable of having most impact on the system. These options are also obtained by means of the Function Keys on the right of the keyboard.

Menu 2 is obtained simply by pressing RETURN from Menu 1.



2.2 THE OPTIONS

FILE The FILE Option is used to change the current file you are working with or to create an entirely new file in the database. Up to fifteen files can be created for each database. Deselect with RETURN. See Section 9.

FORMAT The FORMAT Option is used to define the screen layout of a new file and to revise the screen layout of the current file. Each file may have up to four screens formatted for each of its records. Deselect with 'f1' followed by 'Q'. See Section 4.

BATCH The BATCH Option is used to perform calculations using information from all or selected records in the current file. This is the option you use to carry out UPDATING operations on your files. Deselect with RETURN. See Section 12.

SORT The SORT Option is used to sort all or a selected list of records by fields other than the KEY field (records are already stored in key order). The result of a Sort operation will be a file on disk of the record keys in the order of the Sort parameters. The original file remains unchanged. This list can then be used in the OUTPUT, BATCH, REPORT or SELECT options. Deselect with RETURN. See Section 10.

PROG This is the Option by means of which PROGRAMS can be created and stored on disk. The PROG Option gives Superbase the power of an applications generator, providing a BASIC extended by all of the Superbase commands. Deselect with 'f1' followed by 'Q'. See Programming Section.

MAINTAIN This MAINTAIN Option provides you with a further menu of options to enable you to carry out various utility functions on your files including the EXPORT and IMPORT of data to and from other programs. Deselect with RETURN. See Section 15.

MEMO The MEMO Option enables you to create screens of information which can be accessed later by yourself or by other users of Superbase. In particular you can modify or design HELP SCREENS by means of this option. Deselect with RETURN. See Section 16.

HELP Available from both Main Menus, the HELP Option is used to display a HELP SCREEN designed to give reference information and memory joggers for all of the major facilities offered by Superbase. Deselect with RETURN. See Section 17.

3 COMMAND LINES

3.1 USING DIRECT COMMANDS

Although Superbase has been configured to allow the user to control its many facilities by means of the two Main Menus, it is also possible to bypass the menu system of control and to enter all Superbase commands directly.

Each of the commands included on the Main Menus can be entered in the command area together with a number of supplementary commands provided for the advanced user.

Commands can either be entered individually or joined together with separating colons into a more complex COMMAND LINE.

3.1.1 ENTERING COMMAND LINES

Commands must be typed in lower case (small) letters only, except for the first letter and when abbreviating (see Section 8.10). From either of the Main Menus you can simply type a Command Line such as:

```
'select next:display [Name]'
```

which would obtain the next record in the file and display one field from it, or:

```
'find "chase-list" where [Outstanding] is ">100":output print
the records from "chase-list"'
```

which would print out the records of customers who owe you more than $100.

For the more experienced user of Superbase this method of control is a faster way of using Superbase's facilities since several functions can be carried out consecutively with no further intervention.

3.2 REPEATING COMMANDS

Whenever you execute a Command Line in this way, you can recall it without typing it in again by pressing the left-arrow key at the top left of the keyboard.

This will result in the Command Line last entered being displayed at the

top of the screen, ready for you to execute it again by pressing RETURN or for you to modify it.

The Programming Section describes how to insert line numbers and link Command Lines together to form a PROGRAM.

3.3 THE SYNTAX OF COMMAND LINES

A Command Line consists of a number of CLAUSES each of which is separated by a colon.

Each clause consists of a PRIMARY COMMAND followed by one or more SECONDARY COMMANDS.

The Primary Commands are commands such as those whose names appear on the two Main Menus:

ENTER, SELECT, FIND, OUTPUT, CALC, REPORT, EXECUTE, HELP, FORMAT, BATCH, SORT, PROG, MAINTAIN and MEMO.

Other Primary Commands are provided for more advanced users and are detailed in the Programming Section.

The Secondary Commands differ according to which Primary Command you are using but examples are as follows:

WHERE: Used with FIND as in:

```
'find "listname" where [County] is "=Devon"'
```

FROM: Used with OUTPUT as in:

```
'output the records from "listname"'
```

or used with SORT as in:

```
'sort the records from "listname" on [outstanding]'
```

ALL: Also used with OUTPUT as in:

```
'output all the records'
```

Each of the above Command Lines consist of only a single clause but several clauses can be strung together separated by colons as follows:

```
'find "listname" where [County] is "=Devon":output the records
from "listname"'
```

3.4 FURTHER SECONDARY COMMANDS

Further Secondary Commands and other uses of those above are detailed in the section relating to the Primary Command with which each is associated.

If you are new to Superbase it is recommended that you use the menu system of control until you are familiar with the individual commands.

4 FORMAT

4.1 THE FORMAT OPTION

Superbase stores information in files of records. A file may be a collection of invoices or business cards, or perhaps a collection of recipes for meals, depending on how the file has been set up by means of the FORMAT option.

Each of the individual invoices or business cards or recipes is stored on a separate record in the file, which is put into the file by the use of the ENTER option (see section 5.1).

Before the records can be entered, however, the FORMAT option must be used to design the layout of the records and to determine the type of information they are to contain. Use the 'Maintain Status' Option to obtain a printout of you file layouts in case you need to reformat a file.

The FORMAT option can also be used to change the layout of existing records. If you have created a record format and wish to change it, skip to section 4.1.5.

4.1.1 CREATING A RECORD FORMAT

If you have just started up Superbase and have selected a previously non-existent file, or have set up a new file by means of the FILE option (see section 9.1), you will have been automatically put into the FORMAT option.

You should see a blank screen with the words 'MODE: FORMAT' in the message area at the top left of the screen.

The screen is the 'blank page' on which your screen layout will be designed. Each record may have up to four such screens, each of which can have a different layout and contain different information.

The maximum number of characters per record is 1108. This includes every character visible between field start and end markers for text, constant, key, numeric, and result fields; maximum of 5 characters for date and calendar fields; and a field separator between every 2 fields (i.e. the total number of fields minus 1). Note that a decimal point counts as 1 character, and every numeric and result field has an additional character reserved for the sign. Sign and decimal place character positions are visible between field markers and are counted as part of the field length.

The minimum actual data area per record is 128 bytes. This restriction facilitates Superbase's fast dynamic disk space allocation technique. For further information see the TECHNICAL APPENDIX.

4.1.2 DESIGNING THE RECORD LAYOUT

A record layout consists of two basic types of item: FIELDS, which are the blank slots where items of information will be entered, and DESCRIPTIVE TEXT. You may type any descriptive text onto the screen, but of major importance are the names you decide to give to the fields such as 'ADDRESS' or 'COST' which help make clear what the information in the fields is supposed to be. Up to 1K of descriptive text including lines and borders but excluding the FIELD NAMES themselves, is allowed.

4.1.2.1 FIELD NAMES

These FIELD NAMES are of vital importance since they are the means by which Superbase keeps track of where the data belonging to the file is to be displayed.

More importantly, the field names are your means of referring to the information held in the fields throughout the file.

The field name can be of any length up to twelve characters and is always the last piece of descriptive text to the left of the field, regardless of how many spaces there are between the field and its name. Field names must be on the same line as the field start marker.

The name must all be on one line and must not contain any spaces. If you were to type a field name containing spaces, such as 'TOTAL COST', Superbase would take 'COST' as the field name since it would be the last word to the left of the field. 'TOTAL' would be just a piece of descriptive text. To make 'TOTAL COST' different from 'COST' insert a hyphen or underline between TOTAL and COST.

You may use the same field name more than once in the same record layout. If you do so, the contents of the first occurrence of the field will be duplicated in all subsequent occurrences of the field. Occurrences after the first cannot be edited. Under program control you may therefore ensure that data is protected by creating a field on Screen 0 and making Screen 1 (which would contain an uneditable copy of the original field) the default screen. Moreover, the length of the field and consequently the number of characters displayed may be varied for each copy. This allows you to display only a limited portion of the field contents; but you must always display from the first character rightwards.

Superbase will also ignore the case that the field name is typed in. That is to say that 'ADDRESS' will be treated as the same as 'address' or 'Address'.

4.1.2.2 FIELD TYPES

There are seven types of fields, each corresponding to a different type of information you may wish to store in your records.

Each of these types may also be either a NORMAL Field or a FORCED Field.

A FORCED Field differs from a normal field in that during use of the ENTER option, you are forced to enter some data in forced fields and may not leave them empty.

TEXT FIELDS

TEXT Fields are used for storing items of information like names, addresses or phone numbers, which consist of strings of letters or numbers mixed with letters or other characters such as hyphens.
Examples would be '21 Highview Avenue' or '01-654 8989'.

NUMERIC FIELDS

NUMERIC Fields are for storing numeric information such as prices, or other quantities of any sort and will not accept non-numerical characters.
Examples are '12.75' or '2'.

KEY FIELDS

These are the same as TEXT Fields except that they can be used to locate the record quickly in a KEY SEARCH (see section 6.2.1), and to keep the records sorted in alphabetical order. Every record must contain a KEY Field, which can be anywhere on any screen of the record. A KEY can be up to 30 characters but we recommend short keys for maximum efficiency.

FORCED FIELDS

KEY Fields are always FORCED Fields. Some information must be entered into a forced field - it cannot be left empty.

Any field can be a FORCED Field.

DATE FIELDS

These are for storing dates. They have a fixed length of seven characters; two each for the year and the day and three for the month. Dates must be entered in the form '01Jan82' or 'Oct1883' (either capitals or lower case). Superbase will calculate the day of the week from the date and display it in the message area at the top of the screen. Dates are valid within the range 1 January 1900 and 31 December 1999.

A DATE Field can, however, be set to be eleven characters long so that the day of the week can be displayed in the field itself. In such a case you would enter the date as above and leave the day of the week for Superbase to calculate and display automatically.

CONSTANT FIELDS

CONSTANT Fields are used for holding information which is the same throughout the file of records. An example would be a field for storing the TAX rate in a file such as the Customer Invoice file in Tutorial Two, or a field for storing PI or some other constant in a scientifically oriented application. The initial value to be held in a CONSTANT Field is set during the formatting process and this will appear as a default value in each record during data entry.

If the value in the CONSTANT Field is altered by reformatting the file, then all subsequently created records will display the new value as a default, leaving the old value unchanged in previously existing records. Thus if there were a change of tax rate then new records would contain the new tax rate but records entered while the old rate was in effect would continue to display the original value. To change the constant field in old records use the BATCH option.

When entering data the CONSTANT Field can be overwritten for the current record.

RESULT FIELDS

These are NUMERIC Fields which are to contain a value dependent on the value of the contents of other NUMERIC Fields within the record. A formula such as '[PRICE]*[MARGIN]' is specified, where the square brackets with names enclosed denote the fields referred to by those names and '*' signifies that we want the contents of these two fields to be multiplied together. In such a case the result field would contain the mark-up value for the product.

Superbase will automatically calculate the value to be stored in a RESULT Field without any further intervention by the user. If the values in the fields specified in the formula are altered, Superbase will readjust the value it holds in the RESULT Field.

You may use any constant or BASIC function in your formula, and by using parentheses the formula can be made more complex (e.g. '([PRICE]*0.15) + [PRICE]' which would increase the price by 15%). Square brackets around field

names as in the above example signify the contents of that field.

Note that field names must be full and complete. Up to 79 characters can be used to describe the calculation as it is entered at the top of the screen (the last example above uses 22 characters). The calculation is stored in a compressed form, with a maximum of 30 characters. Each field name takes 2 characters. Spaces are not counted. All arithmetic operators (including opening and closing parentheses) and functions take 1 character each. Field names used in a calculation can be anywhere in the record and on any screen. RESULT Fields can make use of other RESULT Fields, so if 79 characters is insufficient, intermediate RESULT Fields can be created on screens that are not normally viewed.

It is important to bear in mind that you may even use the names of non-numeric fields in a calculation, in which case Superbase will pick up the first item of numeric information in those fields. In a TEXT Field containing the information ' 2 8amp Fuses ' for example, the value '2' would be picked up in a calculation which refers to the contents of that field, as it would if 'Fuses, 2' were contained in the field. If the field contained '8amp fuses, 2' however, the value of the field would be taken as '8'.

It is not until after you have finished designing your record layout and have selected the 'End Format and Store' option that you will be asked to enter the formulae for your RESULT Fields.

CALENDAR FIELDS

CALENDAR Fields store dates like DATE fields, but are also like RESULT fields in that what they contain depends on calculations carried out on other dates within the record.

They could be used for the automatic calculation of regular appointments or of a deadline which is a fixed number of days from another date in the record.

4.1.3 SETTING THE FIELDS

The previous sections have described each of the field types and what they are for. In the sections to follow you will be shown how to set fields of each of these types in your record layouts.

You will see that to set the start of any of the fields, you can use the 'f1' key followed by the first letter of the field type to be set (e.g 'f1' + 'T' for a Text field).

4.1.3.1 SETTING A KEY FIELD

Having typed a field name such as 'NAME' you may set the start of a KEY Field by pressing the 'f1' key followed by the 'K' key.

The message 'Set Key' will appear in the message area at the top left of the screen and a small rectangle will appear just before the flashing cursor. You will also see the number '1' in the right hand message area at the top of the screen.

The small rectangle signifies the position where the KEY Field starts and the number signifies the current length of the field.

After choosing the length of the field (perhaps 15 to allow for names up to fifteen characters long) press the cursor right key at the bottom right of the keyboard to move the cursor along to the end of the key field. The maximum length of a key field is 30 characters.

> Note: The key should be as short as possible while allowing each key to be unique. The shorter your keys the faster your record access times will be, and the less disk space will be occupied by the file index.

You will notice that as the cursor moves along the line the number in the right hand message area will increase to show you the current length of the field.

You may also use the cursor left key to reduce the length of the field.

Once the field is the size you require (check the length count) press the RETURN key to set the end of the field.

Note that field lengths can be easily changed at any time without loss of data, so choosing a field length at this point does not limit you in any way. However, once information has been entered into the Key field, it should not be shortened to less than the maximum used length in any record. REPLACING a record that has had its Key field shortened so that it becomes the same as another record might result in damage to your records.

You will notice a small striped square on the screen just before the cursor. This is the field-end marker, and may be one of two kinds depending on the direction of the stripes.

If the stripes run diagonally from bottom left to top right then the field-end marker signifies the end of a normal field, but if the stripes run diagonally bottom right to top left then the field-end marker signifies the end of a FORCED FIELD.

Placing the cursor over either the field-start or field-end marker will result in the field type being indicated at the top of the screen.

4.1.3.2 SETTING A TEXT FIELD

To set a TEXT Field press the 'f1' key followed by the 'T' key.

You will see another field-start marker appear on the screen, this time a small square. The message 'Set Text' will be displayed in the left-hand message area, and the number '1' on the right to indicate the field length.

Use the cursor right key again to set the length that you require. Check the number of characters with the counter in the right-hand message area and press RETURN to set the end of the field. The maximum length of a Text field is 255 characters, so the field start and end markers may be on different lines. You cannot start a field on one screen and end it on the next.

4.1.3.3 SETTING A DATE FIELD

A DATE Field can be set by positioning the cursor where you want the date to be shown on the record and pressing the 'f1' key followed by the 'D' key.

The word 'Date' will appear in the left-hand message area and the cursor will jump to the end of the field.

Unlike other fields mentioned above, the DATE Field will be set to a fixed length of seven characters, the minimum length for a date field, unless you use the INSERT key to extend it to eleven characters in order to make room for the automatic calculation of the day of the week to be displayed in the field. DATE fields are stored in numeric form using up to a maximum of 5 characters.

4.1.3.4 SETTING A NUMERIC FIELD

A **NUMERIC** Field, e.g. one that is to contain information as to the price of an item, can be set by pressing the 'f1' key followed by the 'N' key.

The message 'Set Numeric' will appear in the message area and a small square preceding a '#' character will appear on the screen just before the cursor to mark the start of the field.

This time there will be two numbers separated by a comma in the right-hand message area; the first representing the number of digits before the decimal point, the second representing the number of digits after the decimal point.

A **NUMERIC** Field may contain up to a maximum of nine digits. There may be up to nine before and up to four after the decimal point. A numeric field that is to contain financial information such as a price will normally have anything up to seven digits before and two digits after the decimal point. Numbers are rounded automatically when necessary, but it should be noted that the results of rounding can be unpredictable when very large numbers are involved, due to limitations in the computer's way of doing arithmetic. However, only very small fractions of a penny (in currency processing) are involved.

Note that although a **NUMERIC** Field does contain a sign for plus or minus (for profit or loss for example) it will not contain a currency sign such as '$' or '£'. The currency sign can be included with the descriptive text immediately before the start of the field.

4.1.3.5 SETTING A RESULT FIELD

To set a **RESULT** Field press the 'f1' key followed by the 'R' key. You will see the message 'Set Result' in the message area at the top of the screen.

The rest of the process of setting a **RESULT** Field is the same as for setting a **NUMERIC** Field above, since the formula for a result calculation is not specified until you have finished formatting the record layout as a whole.

A combined maximum of 32 RESULT, CONSTANT, and CALENDAR fields may be used in a record format.

4.1.3.6 SETTING A CONSTANT FIELD

You can set a **CONSTANT** Field by pressing the 'f1' key followed by the 'C' key. The message 'Set Constant' will be displayed at the top of the screen and you may use the cursor control keys to set the size of the field in the same way as for a TEXT field. The maximum size of a Constant field is 30 characters. The contents of the Constant field are specified at the end of the FORMAT operation.

4.1.3.7 SETTING A CALENDAR FIELD

Set a **CALENDAR** Field by pressing 'f1' followed by SHIFT 'C' key. Apart from the message 'Set Calendar' the process is the same as that for setting a DATE field above. The formula for the Calendar field is specified at the end of the FORMAT operation.

4.1.3.8 SETTING A FORCED FIELD

Any field may be set as a **FORCED** Field simply by pressing SHIFT/RETURN instead of RETURN when setting the end of the field. This will force the user to enter data into the field before continuing to enter data into subsequent fields. Very useful where a field contains vital information which might otherwise be left out.

Note that **KEY** Fields are always **FORCED** Fields. In this case you may press either RETURN or SHIFT/RETURN to set the end of the field. **CONSTANT** Fields are not forced, but are automatically filled with the constant value when records are being added to the file.

4.1.4 FURTHER FORMAT COMMANDS

There are a number of further commands which you can use while creating a record format. They can be used to enhance the appearance of the records and make the records more intelligible by dividing the information in the fields from the descriptive text.

FORMATTING ADDITIONAL SCREENS

When you have finished formatting your current screen you can obtain the next blank screen to format by pressing the 'f1' key followed by the '+' key. Up to four screens are available for each record layout.

RETURNING TO THE PREVIOUS SCREEN

If you are formatting a multiple screen record layout you can return to the previous screen by pressing the 'f1' key followed by the '-' key.

INVERTING A LINE

You can invert a line of the record so that it appears light on dark rather than dark on light. This has the effect of highlighting the line you have inverted.

Position the cursor on the line to be inverted and press the 'f1' key followed by the 'I' key.

INVERTING THE SCREEN

If you prefer, you may invert the whole screen so that the entire record appears light on dark.

Press the 'f1' key followed by the 'S' key.

You may reverse the effect of either Invert Screen or Invert Line by repeating the command.

ERASING A LINE

A whole line can be erased by positioning the cursor on the line to be blanked and pressing the 'f1' key followed by the 'E' key.

Used in this way, the ERASE command will only erase lines of descriptive text such as field names and other text entered onto the screen during formatting. If you want to remove fields from the record you must position the cursor over the field-start marker or the field-end marker before pressing the 'f1' key followed by the 'E' key.

DELETING A LINE

This command differs from the command above in that whereas the latter will replace an original line of descriptive text with a blank line, this command will remove it from the record altogether, shifting all subsequent lines up to fill the gap.

Note that you cannot use this command to delete lines which have fields on them. This is to ensure that you do not remove fields accidentally, but have to consciously go through the procedure described under the ERASE command above.

Just press the 'f1' key followed by the 'INST/DEL' key at the top right of the keyboard.

INSERTING A LINE

Similarly, a new line can be inserted into the record, shifting all subsequent lines down to make room.

Position the cursor at the start of the line before which you want the new line to appear, and press the 'f1' key followed by the 'INST/DEL' key with the 'SHIFT' key held down.

DRAWING A BORDER

You may wish to draw a box around the record or to draw a divider between two parts of the record. Press the 'f1' key followed by the 'B' key. You will then be

asked 'Border Character?'.

Pressing a key at this point will enable you to draw repeated characters of a particular kind simply by using the cursor control keys.

The cursor will leave in its wake a stream of characters of the type you have chosen until a key other than a cursor control key is pressed.

Note that many of the graphics symbols on the front of the keys are available if you hold down the 'SHIFT' or 'CBM' key while selecting the character to be repeated, but you cannot use the symbols used by Superbase as field-start or field-end markers.

Try experimenting with different keys until you find the symbol you wish to use.

To erase a border that has been created in this way, simply select the BORDER option again but choose the space character as the border character and draw a border of spaces.

SCREEN DUMP

You can obtain a paper printout of the screen at any time by holding down the CONTROL key while pressing the 'P' key.

QUIT FORMAT

If you wish to escape from the Format Option at any point and return to Menu 1, you can do so by pressing the 'f1' key followed by the 'Q' key.

You may wish to do this if you have entered the Format Option by mistake. Any changes made prior to this command will not be saved.

CLEAR FORMAT

If you are totally dissatisfied with the format you have designed, you can clear the screen to start again by pressing the 'f1' key and then the 'CLR' key at the top left of the keyboard.

Note that only the current screen is cleared if you are working with a multi-screen file.

This is another key which requires the SHIFT key to be held down.

END FORMAT AND STORE

When you have finished formatting your record you can store the screen layout by pressing the 'f1' key followed by the 'STOP' key.

You will notice each of the field-start and field-end markers being replaced by angle brackets.

If you have included **RESULT** or **CALENDAR** Fields in your record format then you will be asked:

'Enter Calculation?'

Type in the formula you require and press RETURN (see section 4.1.3.1 above for details of formulae in **RESULT** Fields).

You will also be asked to specify the contents of any **CONSTANT** Fields at this point.

DUPLICATE KEYS

After specifying the calculations and constants you will be asked:

'Allow Duplicate Keys?'

If your application can be set up using a UNIQUE KEY then type 'N' for 'no' in response to this question, otherwise type 'Y' for 'yes'.

A duplicate key file is a file in which more than one record can have the same KEY. In a file such as the Address Book file in Tutorial One, where the Key Field is NAME, there could be two separate 'John Smiths'.

The disadvantage of duplicate keys lies in accessing records with identical keys from a key list. Although a duplicate key such as 'John Smith' will appear on the key list as many times as there are records with that key, when the list is used only the first actual record will be accessed, over and over again, for each identical key encountered. This obliges you to devise a program using the SELECT NEXT command to obtain the second and subsequent 'John Smith' records.

It is recommended that you normally use UNIQUE KEYS. Only use DUPLICATE KEYS where there is an overriding reason for doing so. One good reason might be to have the facility to obtain the most

recently created record of a number of records sharing the same key, where records are records of orders, for example. If Superbase has to add a duplicate key, it inserts the key in the key index before any existing identical keys. This means that every search for such a key will retrieve according to the principle of 'last in, first out.'

4.1.5 EDITING AN EXISTING RECORD FORMAT

If at any time you wish to change the format of one of your files you may do so without losing any data. You should always do a STATUS command on the file to print out the format characteristics including any formulae before embarking on a file reformat operation. See Reference Section 15.1.1

FIELDS MAY ONLY BE REMOVED FROM OR ADDED TO THE END OF THE RECORD.

You can change the length of the fields or change the descriptive text. The only thing that must remain the same if you are to preserve the link between the data and the record layout is the name of each of the fields in which the data is to be displayed. Although field types may be changed, this should be done with caution as you may inadvertently cause damage to your data.

From Menu 2, obtain the FORMAT option by using the 'f2' key on the right of the keyboard. Hold down the SHIFT key and press the key marked 'f1' on top and 'f2' at the front.

Move through the existing record format using the INST/DEL key to insert or remove spaces inside field markers. Add lines, borders, etc. as required. Change field types with the following rules and guidelines in mind.

1 If a date or calendar field is converted to a non-date or calendar field, the date that was in it will be displayed as a number.

2 If you change a field to become a result or calendar field, the formula for the next following result or calendar field will be displayed for editing at the end of the operation. You must overtype the display with the new formula, and re-enter the old formula into the proper result field.

3 If you change a text field to a numeric field then any text data in that field will be lost when the field is subsequently edited.

4 If you specify another field as the key field, the current key will be re-assigned to that field and whatever was in that field will be lost. This procedure is NOT recommended!

5 If you change a replica (i.e. copied) field to a unique field, you will create a new field. This will not affect the order of fields within the record. The new field will have no data in it. Likewise, converting unique fields to replica fields is permissible, but will result in the data from the original of the replica being displayed in that field.

Full examples of setting up record layouts are given in Tutorials One and Two.

4.1.6 TRANSFERRING AN EXISTING RECORD FORMAT BETWEEN DATABASES

If you wish to create a file format in either the current or another Superbase database that is the same or nearly the same as an already existing one, there is an easy way of doing so.

Use the MAINTAIN OTHER Option to copy the file format information to a file with a new name (see Reference Section 15.6). The records themselves cannot be copied as they are stored in the Database file itself, which is listed in the disk directory in upper case letters. Once you have created the new format file, you can enter its name after selecting the FILE Option, either in the current or the new database.

4.2 SUMMARY

The Format Option is used to set up screen layouts for the records in a file, or to modify already existing screen layouts.

You can obtain the FORMAT option to revise a screen layout by pressing the 'f2' key from Menu 2.

You are automatically put into the FORMAT option whenever you select an as yet non-existent file from the Database Catalog, either at start-up or while using the FILE option.

Sets a Key Field.

Sets a Text field.

Sets a Date field.

Sets a Numeric field.

Sets a Result field.

Sets a Constant field.

Sets a Calendar field.

Gives you the next screen to format for this file (up to four, numbered 0 to 3).

Takes you back to the previous screen.

Inverts the line of the screen where the cursor is currently positioned.

Inverts the whole screen.

Erases a line of descriptive text where the cursor is currently positioned.

Erases the field which has the cursor positioned over its field start or field end marker.

Deletes the line the cursor is currently on and moves up the subsequent text to fill the gap left behind.

Inserts a line just before the current cursor position and pushes the subsequent text down to make room.

Enables the cursor control keys to be used to draw a line or border of characters of your choice.

SHIFT for f2 f4 f6 f8

REFERENCE - FORMAT

Causes whatever is currently on the screen to be printed out.

Enables you to quit the Format option and return to Menu 1. The Format is abandoned.

Clears the whole format on the current screen to enable you to start again.

Ends the formatting process, asks for calculations and constants, and stores the record format.

5 ENTER

5.1 ENTERING INFORMATION

Once you have created a file (see section 4) you can enter information into it.

With the ENTER option, you fill in a blank record with the information you want to keep, then add that filled-in record to the file.

5.1.1 OBTAINING THE ENTER OPTION

From Menu 1 press the f1 key on the right of the keyboard.

You should see the blank record from the file selected appear on the screen.

Superbase uses the message area at the top of the screen to tell you:

MODE: ENTER	that you are in the ENTER option
#1 k	that the field of the record you are about to enter information into is the first field, and is a KEY FIELD.

Note that the first field of the record need not have been a KEY FIELD. Every record must contain a key field but it can be in any position in the record.

You are now ready to fill in each item with the desired information. All characters are valid except double quotation marks (").

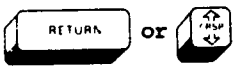
5.1.2 FILLING IN THE BLANK RECORD

The following control keys allow you to move the cursor so that you can enter information where you want:

Moves the cursor to the right unless the cursor is at the end of the field, in which case it moves the cursor to the next field.

REFERENCE - ENTER

Moves the cursor to the left unless the cursor is at the beginning of the field, in which case it moves the cursor to the previous field.



Moves the cursor directly to the next field.

Moves the cursor directly to the previous field.

Sends the cursor to the first field in the default screen. Any of the four screens can be set as the default screen by typing 'Screen n' from the Main Menu, where n is the number of the screen wanted as the default screen. This screen will be the first to be displayed when ENTER is selected.

Clears all of the fields in the record and sends the cursor to the first field of the default screen.

Enables you to exit from the ENTRY option without any information being stored. This is useful if you have selected the ENTRY option by mistake.

Dumps the current screen to the printer to provide you with a hard-copy print out of the screen layout or the record you are currently entering.

Moves the cursor directly to the next field unless the cursor is in the last field of the last screen, in which case it displays 'Press Return to Store' in the message area. Pressing RETURN again at this point stores the record, along with the information you have entered, into the file.

Note that if the current file is NOT a Duplicate Key File (see section 4.1.5) and the KEY FIELD contains information identical to that in an already existing record, the message 'Key Already Exists' will be displayed in the message area together with 'Press Return to Continue'. The record you have just defined will NOT be stored. You will remain in the ENTER option so that you can type a new key.

To change an existing record you should use the REPLACE function of the SELECT option (see section 6.2.10).

Displays 'Press Return' in the message area. Pressing RETURN will then store the record in the file. This command can be entered wherever the cursor is at the time. Cancel it by pressing any key other than RETURN.

5.1.3 FORCED FIELDS

If the cursor is in a field which has been set up as a FORCED FIELD (see section 4.1.3.1), you will be prevented from storing the record until some information has been typed in all FORCED FIELDS.

The message 'Forced Field: Please Enter Data' will be displayed in the message area at the top of the screen and the cursor is placed in the first FORCED FIELD.

5.1.4 DATE FIELDS

In a field which was defined as a DATE Field during record formatting (see section 4.1.3.1), Superbase will only accept entries of the form '18oct93' or 'oct1883' (capitals or lower case). Superbase accepts single character dates so it is unnecessary to enter 04oct83 as 4oct83 will be accepted.

Attempting to enter a sequence of characters of any other kind will result in the message 'Invalid Date' being displayed in the message area, and the cursor will be prevented from moving to another field until a valid date has been entered.

Once a valid date has been entered, Superbase will calculate which day of the week corresponds to that date and will display that day in the message area at the top left center of the screen.

If the DATE Field is long enough (11 or more characters) Superbase will also display the day of the week in the field itself.

To enter a date a designated number of days forward, merely add days to the date and Superbase will calculate the valid date. For example, if the date is 12Jan and you require 20 days on, type in 32Jan.

See the DATE command in Programming Section 2. Single keystroke input of a pre-assigned date may be achieved under Program control.

Full examples of the use of the ENTER Option are given in Tutorials One and Two.

5.1.5 MODIFYING NUMERIC FIELDS

There is an easy way of entering multiples of numbers into **NUMERIC** Fields. Suppose you know that your customer has bought 17 items at $37.95. First enter 37.95 into the **NUMERIC** Field. Then place the cursor at the beginning of the field and type in '17*' and press RETURN. The value stored in the field will be the result of multiplying 37.95 by 17.

You may modify **NUMERIC** Fields in this way using any of the arithmetic operators; '+', '-', '*' or '/' provided a space is available in the field for the additional character.

If you want the number and operator to be after the field contents, as in '/6', you can use the DELETE key to pull the field contents over to the left so that you can fit in your number and operator.

This facility for modifying the contents of a **NUMERIC** Field is also available with the ADD and REPLACE Options in the SELECT Option.

5.3 SUMMARY

The ENTER option is used to store information into new records in your file.

Selects the ENTER option from the Main Menu.

Moves the cursor to the right within a field, or if at the end of a field, moves the cursor to the next field.

Moves the cursor to the left within a field, or if at the beginning of a field, moves the cursor to the previous field.

Moves the cursor directly to the next field.

Moves the cursor directly to the previous field.

Moves the cursor directly to the next field, or if in the last field, when pressed again will store the record in the file.

With the cursor in any position, displays 'PRESS RETURN' message. Pressing RETURN again stores the record in the file.

Moves the cursor to the first field of the default screen.

Clears all of the fields in the record and sends the cursor to the first field of the default screen.

Exits from the ENTER option without any data being entered.

Prints the current screen.

6 SELECT

6.1 SELECTING A RECORD

The SELECT Option provides you with a number of ways of obtaining particular records from your file.

Note that records are stored in the file alphabetically according to the contents of the KEY Field of each record (see section 4.1.2.2). Numbers are treated as coming 'alphabetically' before letters.

6.1.1 OBTAINING THE SELECT OPTION

To obtain the SELECT Option from Menu 1, use the 'f2' key at the top right of the keyboard. This is the 'f1' key with the SHIFT key held down.

You should now see a further Menu, the SELECT Menu, with a number of sub-options listed on the screen. To return to this Menu after a SELECT operation, press RETURN. Press RETURN again for Menu 1.

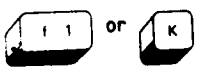
```
mode : Select Menu

 f1  Key
 f2  Current
 f3  Next
 f4  Last
 f5  Previous
 f6  First
 f7  Match
 f8  Output
 A   Add
 R   Replace
 D   Delete
```

6.2 THE SELECT OPTION CONTROL KEYS

The various choices available to you at this point, and the control keys used to obtain them, are as follows. From the SELECT Menu or from any SELECT option use the function key selection or the initial character of the SELECT option.

6.2.1 KEY



This command enables you to search your file for a record with particular information in the KEY Field. On selection you will be asked: 'Key?'

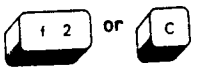
Type in the information you think is held in the KEY Field of the record you want and press RETURN. You may enter the full or partial contents of a Key Field for a full or partial match.

Superbase will then display either the record whose key matches your entry exactly, or the record that contains the nearest key to the characters entered.

If a record is found with a key that starts with the same characters as those you entered, but has more characters, then the message 'Partial Match' will be displayed.

If no record has a KEY starting with your specified characters, then the message 'Key Not Found' will be displayed together with the record which has the alphabetically closest KEY.

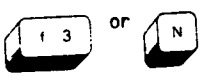
6.2.2 CURRENT



This command obtains the record you are currently working on in this file.

Superbase will remember the CURRENT record for up to the last three files used. This can be used in multifile programming situations.

6.2.3 NEXT



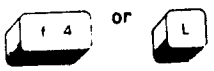
If you have just been using a record or have just viewed a record using one of tne other commands, this command obtains the NEXT record in the file.

This will be the next record alphabetically according to the key

field. Use 'M' to obtain next 'Matching' record (see Section 6.2.7).

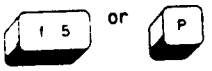
If you have not yet accessed any records in this file, then attempting to use 'Next' will give you the first record in the file.

6.2.4 LAST



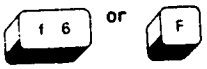
This command obtains the **LAST** record in the file. In a file of invoices with the invoice number as the key, the LAST command would display the invoice with the highest number. This would be the most recent invoice stored.

6.2.5 PREVIOUS



After viewing a record you may wish to see the **PREVIOUS** record in the file, in which case use this command.

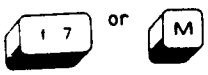
6.2.6 FIRST



Use this command to obtain the **FIRST** record in your file.

6.2.7 MATCH

The **MATCH** Command provides you with a slower but much more flexible method of searching your file for particular records.



You may use this option to view selected records based on a wide range of criteria in any field or combination of fields.

SUMMARY OF SELECTION CRITERIA

Note that "text" can be letters or dates, as appropriate.

Exact match	"=text"	"=number"
Sliding match within field	"text"	
Exact exclusive match (not equal to)	"#text"	"#number"
Sliding match from field onwards	"text-"	
Terminate the sliding match	"*"	
Match greater than field content	">text"	">number"
Match less than field content	"<text"	"<number"
Match alternatives	"t1/2/etc"	"n1/n2/etc"
Match range of values	">t1&<t2"	">n1&<n2"
Match partial text	"=text*"	
Match with 'wild' character	"=text??"	
Request match in command area	"<-" (back-arrow)	

You can specify that the records selected have particular values in specified fields, or that they have that value in a field within a particular range of fields.

It is possible to require that they have EITHER of two values in a specific field or range of fields, or that a field or one of a range of fields contains a particular string of characters amongst others.

You can also specify that a particular field or one of a range of fields holds a value within certain limits, such as between 100 and 500.

Facilities such as these, alone or in combination, ensure that even in a file of thousands of records you can select any particular record or group of records you want with ease.

RECORD TEMPLATE

On entering the MATCH option you will be asked: 'Select Match Data'. Superbase will display a blank record on the screen with the cursor in the first field. This is the 'Record Template' into which you enter the criteria to be matched during the search, which begins when you have finished entering criteria.

Using the cursor control keys and RETURN to move from field to field, you may now enter the specification that will determine which record or records are to be selected and displayed.

ENTERING CRITERIA FOR MATCHING

Each field may either be left blank or may have some characters (numbers, letters, or other characters) typed into it. These characters are called the CRITERION for that field. The criterion typed into a field will consist of a string of characters (e.g. 'London') optionally preceded by an operator (e.g. '=London'). The operators available are '=' (equal to), '<' (less than), '>' (greater than) and '#' (not equal to).

If no operator is provided then SUPERBASE will simply look for the string of characters ANYWHERE in the specified field. This is called a SLIDING MATCH and is detailed below.

If a field is not left blank then the specification entered into it will be compared with the contents of each record and only those records which match that specification will be selected.

For example, If '=Jones' is typed into the 'Name' field of an Invoice record, then only those Invoice records pertaining to Jones will be selected.

Similarly, if the specification consists of '=Jones' in the 'Name' field and '>500' in the 'Amount Due' field then only those of Jones's invoices for more than $500 will be selected.

STARTING THE SEARCH

Once you have finished entering your specifications into the fields you can store the match criteria and start the search by holding down SHIFT and pressing RETURN. If the cursor is in the last field of the record then just RETURN will do.

The first such record will be displayed and any others can be obtained by pressing 'M' for 'Match' repeatedly.

Each time 'M' is pressed, the next record in the file which matches the specification will be displayed.

INTERRUPTING AND RESUMING THE SEARCH

At any time you may instead interrupt the display of selected records by pressing 'N' for 'Next' or 'P' for 'Previous', in which case the next or previous record alphabetically according to the key field will be displayed, whether or not it matches your specification.

Pressing the 'M' key again will resume the display of records matching the specified criteria.

In this way you can view the records selected by your 'Match' criteria and pause occasionally to browse through neighboring records in the file.

You may even use the ADD or REPLACE commands to modify the file and then continue viewing the selected records where you left off.

ENDING THE SEARCH

To cancel the current MATCH operation in order to set up new match criteria, use the LAST command to jump to the end of the file. You may then use MATCH to enter new specifications.

6.2.7.1 SLIDING MATCH

You may search for a string of characters anywhere in a field by omitting the '=' symbol or other operator in the match specification.

This is known as a 'Sliding Match'.

Whereas '=Desk' would only select records with precisely the word 'Desk' in the appropriate field, 'Desk' would also select records with 'Large Desk', 'Pine-desk', 'desktop' and so on in the specified field.

You can also combine a sliding match criterion with other criteria using operators such as '#', '=' or '>'.

6.2.7.2 FIELD-INDEPENDENT MATCHES

Another facility offered by the MATCH Option is a field-independent match.

Place the cursor in any field and type the specification followed by a hyphen (e.g. 'London-' or '50-'). The records selected will be those with the specified criterion occurring in any subsequent field.

As described above, the field-independent match will look for your specified criterion in all of the fields including and following the field where the specification was entered. It is also possible to restrict the range of fields to be included by typing the FIELD-INDEPENDENT MATCH DELIMITER symbol '*'. This will exclude the field it occurs in and all subsequent fields from the fields to be searched, thus speeding up the search. Also, if you place a field-independent match in field 1 and an exact match in field 4, the search will be canceled for fields after the exact match.

This facility may also be combined with other types of match criteria. Other criteria must be before the field-independent match and after the delimiter '*'.

6.2.7.3 THE 'AND' OPERATOR AND THE 'OR' OPERATOR

You may also use the operator '&' (AND) to specify a range of values, and the operator '/' (OR) to specify more than one possible criterion within a given field.

For instance, the specification '=London/=Birmingham' would select those records with EITHER 'London' OR 'Birmingham' appearing in the chosen field.

Similarly, the specification '>100&<150' would select only those records with a value in the given field above 100 AND below 150.

6.2.7.4 PATTERN MATCHING

Pattern matching may also be used within a 'Match' specification. The '*' symbol indicates that any string of characters beginning with the characters preceding it will be accepted. Pattern matching characters should be used in conjunction with the '=' operator, as sliding matches are available for wider ranging matches.

Thus '=Sm*' would select any record with 'Smith', 'Smythe', 'Smullyan' etc. appearing in the chosen field.

Note that the '*' symbol can only be put after a string of characters and must have no other characters following it.

Similarly the '?' symbol can be used as a 'wildcard' character so that 'Hutch?ns' would select both 'Hutchens' and 'Hutchins'.

These pattern matching symbols are very useful if you cannot remember how a name is spelled, or if you suspect that it was spelled incorrectly on entry, but they are also useful if you want to ignore characters in a string for the purposes of the match.

6.2.7.5 USING THE BACK ARROW

If you wish to enter a criterion that is longer than the space allowed for it on the screen, such as the specification of combinations and alternatives, the back-arrow '<-' provides a way of doing so.

Instead of entering the specification, place a back-arrow in the first character position of the field. Press RETURN and continue with the other criteria if any.

When you have finished specifying criteria in the record template, Superbase will prompt you in the command area to enter the criterion for the field with the back-arrow in it. Only when all such fields have had their specifications entered will the search begin.

6.2.7.6 SUMMARY

These various types of Match Criteria, used individually or in combination, add up to a powerful facility for selecting records from your files. You will see in Section 7 that groups of records selected in this way can be indexed by a list called a KEY LIST which can be stored for future use.

6.2.8 OUTPUT

The OUTPUT command available from the SELECT Option provides a display of the field names and field contents of the CURRENT record, either on the screen or to the printer. To output to the screen use 'display' and to output to the printer use 'print'.

The contents of the record are displayed across the page. The full field sizes are used and information wraps over onto following lines.

You can also change the format of the output from ACROSS TO DOWN by typing 'down' from either Main Menu. The two commands can be typed in together as in 'display down' to display each field on a new line.

The field names are displayed down the left hand side of the screen if the display is DOWN and only if all fields are output, with the contents of each field to the right of the field name.

If you want the output to be directed to the printer instead of the screen you must first change the direction of output by typing 'print' from either Main Menu.

Whichever of the commands 'display', 'print', 'down' or 'across' were last used will remain in effect until the converse commands are used.

6.2.9 ADD

The ADD command is for adding a new record based on an existing record. The new record must have a different key field and may also be changed in any other fields.

This is useful if you want to enter records which have much information in common with an existing record. It allows you to avoid typing all the information in again.

Using the KEY, NEXT, PREVIOUS or other command, select the record you wish to base the new record on and then use the ADD command to create it.

Note that unless you are working with a Duplicate Key File, the key field of the new record created by means of this command must differ from that of the record on which it was based. If you have not changed the key field, Superbase will display the message 'Key Exists' and will return the cursor to the key field.

6.2.10 REPLACE

REPLACE is the facility you use to edit the records in your files. It is like the ADD command except that instead of producing a new record based on a record you have selected, it will instead replace the selected record with the modified version you have created.

Whereas the ADD command requires that the key field is changed so that the new record can be distinguished from the record it was based on, the REPLACE command requires that the key field remains the same as in the record to be replaced.

See Section 5.1.5 for discussion of an easy method of modifying the contents of numeric fields.

6.2.11 DELETE

This command is used to delete the record that is currently selected from the file.

You will be asked: 'Confirm Deleted Record' so that you can change your mind before any harm is done. Type 'N' for 'no' if you have made a mistake, otherwise type 'Y' for 'yes' and the record will be permanently removed from the file.

6.3 ALTERNATIVE WAYS TO EXECUTE THE SELECT COMMANDS

Note that each of these commands can also be chosen by typing the first letter of the command (e.g. for 'First' type 'F'). A list of these letters is provided in the message area above the main screen throughout the SELECT Option.

It is also possible to bypass the Select Menu by typing one or more of the Select commands directly from either of the Main Menus. The command line you type (e.g. 'Select First') will appear in the message area above the screen. Just press RETURN and the command line will be executed.

You can even join commands together, separating them with colons (':') to create longer commands such as 'select match where [Name] is "=Jones":select next:display [Name].

Superbase will execute these commands one after the other without any pause between them. (See the Additional Commands Section for an explanation of the 'where' command.) If there is an error in your command line, Superbase will display an error message and stop. You can recall the line and make changes to it by pressing the back arrow key. More details about these COMMAND LINES, as they are called, are given in Section 3 and The Programming Section.

7 FIND

7.1 USING THE FIND OPTION

The FIND command enables you to set up a list of the KEYS of a selected group of records which can then be used with the SORT, OUTPUT, BATCH, REPORT and SELECT commands to access and process just those records on the list.

The process of specifying which records are to be included in the KEY LIST is exactly the same as in the MATCH command of the SELECT option (see section 6.2.7).

7.1.1 OBTAINING THE FIND OPTION

To obtain the FIND option press the 'f3' key from Menu 1.

You will see a blank record screen as in the MATCH option, ready for you to enter the match criteria governing the selection of the records to be included in the list.

7.2 THE DEFAULT LIST

Once you have entered your match criteria as detailed in section 6.2.7, the message 'Processing' will be displayed at the top of the screen while Superbase searches through the file for the records specified, adding the keys of all matching records to its KEY LIST. You will then be returned to Menu 1.

Unless you specify a name for the KEY LIST as will be detailed below, Superbase will give the list the default name "hlist". Like all lists, "hlist" is automatically stored on disk. Every time a new "hlist" is created, the old one is lost.

The reason for the leading 'h' is to allow you to display the list by using the HELP command. As explained in Section 16, all files with names preceded by an 'h' are treated as HELP SCREENS and so can be viewed by means of the HELP option.

To view the current "hlist", obtain the HELP option and type 'list' when asked to specify which help you require.

Whatever Key List you last created by the FIND command will remain the current "hlist" until you create another.

7.3 NAMING AND STORING A KEY LIST

Using the default Key List is convenient for constructing temporary lists of records for various purposes, but there will be many Key Lists that you will want to keep to use again and again.

You may, for example, want a permanent list of customers who have bought a particular product from you, or a list of products with a particular discount rate.

For purposes such as these you need to be able to store a Key List on disk with the appropriate file. In order to do this, all that is required is to give your Key List a name other than "hlist". The maximum length of a list name is 16 characters.

From either Main Menu, type the COMMAND LINE **'find "listname"'**, where the listname can be any name not already used as the name of a Superbase file, and must be enclosed in double quotation marks.

Do not use the name of a database. If you use the name of an existing list or file, it will be overwritten by the new list.

To avoid confusion it is a good idea to call all of your Key Lists by a name ending in 'list'. Examples would be "update-list", "10%d-list" and so on.

To append to an existing list, use the form: **'find "update-list,a"'**. This is useful when you already have a list (e.g. last month's orders) and want to add to it without repeating the original processing.

After typing **'find "listname"'** from either Menu, you will be presented with the blank record template for entering your match criteria just as you are when you press 'f3' to obtain the FIND command from the list of options on Menu 1.

7.3.1 BYPASSING THE RECORD TEMPLATE

You can even bypass the record template altogether by entering your match criteria in a COMMAND LINE from either Menu. See Section 3 and the Programming Section for further details of command lines.

In a FIND command line you can specify several match criteria one after the other, using the WHERE secondary command to indicate the start of the list of criteria sought for the record. Instead of placing the operator (if any) and text or number value inside the field angle brackets, you must place it inside double quotation marks. The rules for matching are the same as when using the record template.

You can if you wish place the back-arrow "<-" inside quotation marks instead of a value. This will produce prompts for any such fields before the search begins, which is useful when long strings of characters must be entered. When done under program control this

An example would be:

```
'find "10%d-list" where [discount] is "=10%" [product] is "<-"'
```

This is an optional method of specifying the match criteria - you can always use the record template if you wish, but the Key List must be given its name by means of the direct command 'Find "listname"' if it is to be stored on disk.

7.4 USING THE KEY LIST

Once a Key List has been created it can be put to use in a number of ways. You might want to obtain printed copy from the records in the list. Either obtain the OUTPUT option (see section 8) and in response to its prompt type 'the records from "listname"' (don't forget that "listname" here can be "hlist" if you want), or you can type the following Command Line directly from one of the Main Menus:

```
'output the records from "listname"'
```

This is only one limited example of the use of a command using a Key List. Details of its use with the SORT, BATCH, REPORT and SELECT commands are given in the sections relating to those options.

Because the old "hlist" is overwritten every time a new default list is written, you will want to rename some "hlists" immediately after creation. Select the OTHER Option from the Maintain Menu, and type:

```
'r0:newlistname=0:hlist
```

See Section 15.6.

8 OUTPUT

8.1 OUTPUTTING INFORMATION FROM FILES

The OUTPUT option is used to display or print information from all or selected records in the current file. Selected or all fields, descriptive text, calculations, and BASIC variables can be output. Special commands can be used to format the output fields. Output commands, like other Superbase commands, may be abbreviated.

8.1.1 OBTAINING THE OUTPUT OPTION

From Menu 1, press the f4 key. This is the key marked f3 on top and f4 on the front pressed while the Shift key is held down. You will be prompted with:

'Enter: all/from "list" (item list....)'

You are being asked whether you want the output from ALL of the records or just those from a predefined KEY LIST, and which fields you want to output. The '(item list....)' signifies that field names and/or variables may be included in the command.

8.1.2 OUTPUT WITH OR WITHOUT FUNCTION KEY

The form of the command is different according to whether it is executed from the menu or typed on the command line.

If you press the f4 key you will be prompted as above. But if you type the command, you must supply the word 'output' at the beginning of the line. If you use the back-arrow key to recall an 'output' command line that was done with the function key, you must insert the word 'output' to make it execute properly.

'Print' and 'display' do not work as first words in the command if it contains any 'output' syntax, i.e. 'all', 'from "listname"', 'the records', etc.

The examples in this section usually assume that you have pressed a function key, so you may need to insert the word 'output' before they will work.

8.2 DISPLAYING ALL RECORDS

The simplest response to the output prompt is 'all'. Field names followed by their contents are displayed one after another across the screen with data wrapping over the ends of lines.

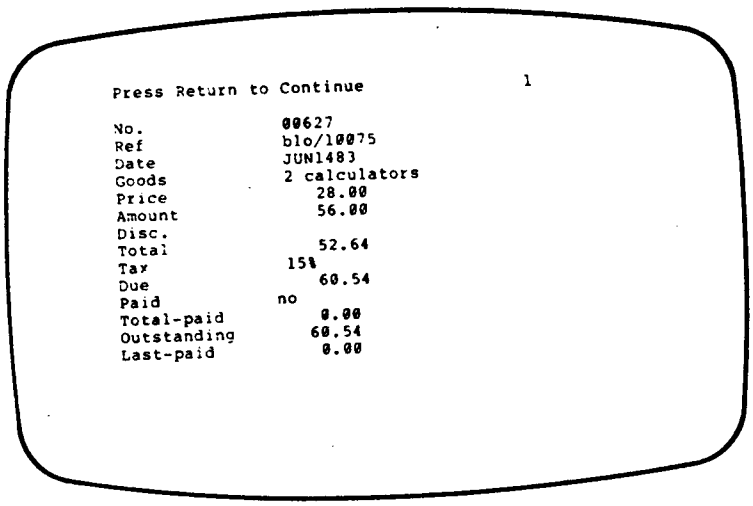
Pressing RETURN displays the next screenful of records from the file. Display of records can continue until the end of the file is reached.

Responding to the output prompt with:

'all the records down'

will result in information from the first record in the file being displayed down the screen with the field names on the left of the screen and the field contents to their right. The message 'Press Return to Continue' will be displayed on the command line.

Pressing RETURN results in a similar display of the next record in the file, record by record until the end of the file is reached.



8.3 DISPLAYING FROM A KEY LIST

Alternatively you could respond to the prompt with:

'the records from "listname"'

where "listname" is the name of any KEY LIST you have created by means of the FIND command, including "hlist" (see section 7).

The display would be the same as with the 'all' command, but would be of only those records whose keys appear on the Key List you have specified.

8.4 PRINTING OUTPUT

To obtain your output on the printer rather than the screen, the command is:

'print all the records'

or

'print the records from "listname"'

The converse of the 'print' command is the 'display' command whic redirects the output to the screen, as in:

'display all the records'

The command last used, 'print' or 'display' will remain in effect unti the direction of output is changed by the converse command.

8.5 OUTPUTTING SELECTED FIELDS

The commands above output information from all of the fields in each record. You can restrict the amount of information to be displayed or printed to one or more fields by responding to the output prompt:

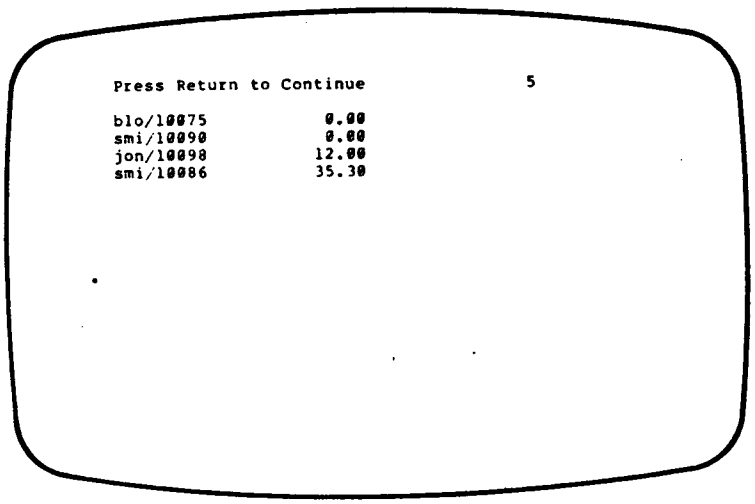
'all the records [Ref]'

or

'all the records [Ref] [Total-paid]'

or any combination of fields in the records, where each field name is enclosed in square brackets.

The result is as in the following screen:



8.6 OUTPUT ACROSS OR DOWN

If your display is 'across' the screen, a space will be left between each selected field. If output is 'down' each selected field will be on a new line.

To display across respond to the prompt with:

'display across all the records [Ref] [Total-paid]'

After filling the screen with the information from each record displayed horizontally, the message 'End of Page' will appear at the top of the screen.

If more fields were displayed than could fit across the screen, the information would wrap around the edge of the screen. This can be avoided by use of the formatting commands detailed below.

To reverse the 'across' command you can use the 'down' command as in:

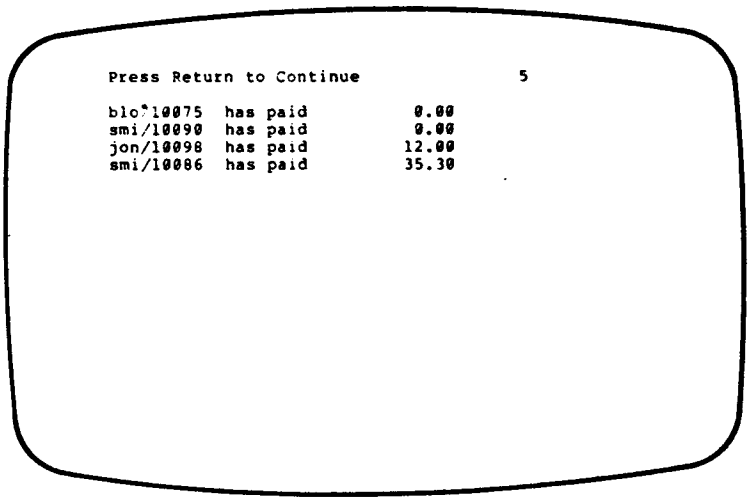
'display down all [Name] [paid] [item]'

Just like the 'print' and 'display' commands above, whichever of the 'across' or 'down' commands was used last will remain in force until the converse command is used. The default on start up is 'across'.

8.7 OUTPUTTING EXTRA TEXT WITH INFORMATION FROM RECORDS

You may also include strings of text with your output as in:

```
'display across all [Ref] "has paid" [Total-paid]'
```



Note that if formulae and BASIC variables are to be output they should not be enclosed in quotation marks like text strings.

8.8 OUTPUTTING CALCULATIONS ON INFORMATION IN FIELDS

The information displayed or printed need not be restricted to how it appears in the records since you can specify calculations to be performed on the fields before they are displayed.

Suppose that you wanted to output a list of items showing a 10% price increase. You could respond to the output prompt with:

```
'across all [item] "New price is" 1.1*[price]'
```

Any calculation on the field contents can be used including calculations using BASIC functions. More details on performing calculations on the fields is provided in the section on the CALC command in section 11.

BASIC variables can also be output in addition to, the fields, calculations and text strings discussed in this section. This is detailed in the Programming Section.

8.9 FORMATTING OUTPUT INFORMATION

Two commands are available to improve the appearance of the output on screen or in your printed copy: the TRUNCATOR COMMAND '&' and the POSITION COMMAND '@'.

8.9.1 TRUNCATING OUTPUT

Text fields will be printed or displayed with the length specified when the fields were set by means of the FORMAT Option (see section 4) unless the TRUNCATOR COMMAND is used.

A text field such as 'Item' may have been set to eighteen characters long to allow for big product names, but the product names included in the Key List you are using may not actually be more than ten characters long.

This would mean that when these product names are output, the extra spaces after the product name would be included unnecessarily. To avoid this you can include the truncator symbol with your response to the output prompt as in:

'display across all &[Item] [price]'

This would chop off all of the trailing spaces from [Item].

It is also possible to truncate the output to a particular length. Typing a number after the truncator symbol will truncate the contents of the field concerned to the number of characters specified. For example

'print all &6[Item] [price]'

would print just the first six characters from the 'Item' field.

The contents of numeric fields can also be truncated, but two numbers separated by commas must be included after the truncator symbol, one for the number of digits before the decimal point and one for the number of digits after the decimal point. The default numeric format is 10 positions before and 2 after the decimal point.

You can mix truncation commands for text and numeric items, as each truncation command applies to the NEXT appropriate item.

Truncation forces rounding. If the field contains a number with more digits after the decimal point than specified by the truncator command the number will be rounded up. For instance, if the price of an item were '75.87' and the output specification were:

'all [Item] &2,1[price]'

Then the price would be output as '75.9'.

You must take care, however, not to specify a truncation of a

numeric field which has fewer digits BEFORE the decimal point than a number to be output. If you do, the number will be printed or displayed as a string of '#' symbols to indicate that the number overflowed the format you specified for it.

The command &0,x will suppress leading spaces in the next numeric item, causing left justification with one space for the sign (only the '-' sign is printed).

8.9.2 POSITIONING OUTPUT

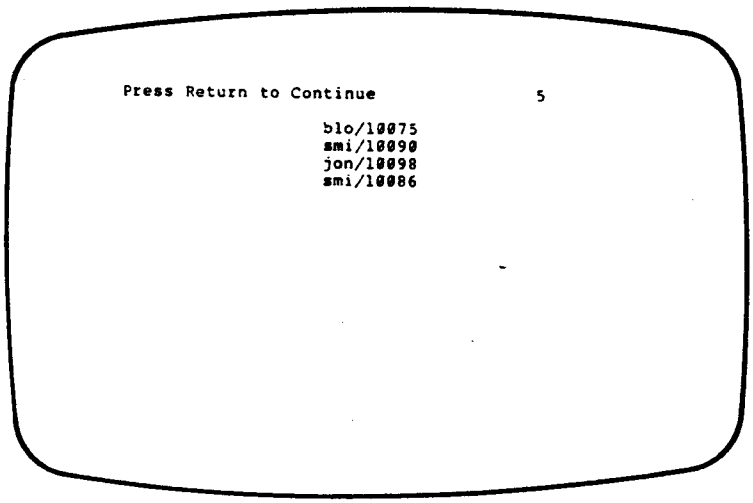
The second formatting command available for use with OUTPUT is the POSITION COMMAND.

This command is used to position your output on the screen or printed page and must have either one or two numbers following it, the first for the COLUMN and the second for the LINE.

For instance, the command

```
'all the records @20[Ref]'
```

will position the contents of the 'Ref' field at column 20, halfway across the screen, as in the following diagram:



The command applies only to the NEXT item. Following items without their own positioning commands will appear in sequence either across or down as appropriate. To achieve column and line location use a command such as

```
'the records from "hlist" @10,15[Ref]'
```

which would position the output at column 10 and 15 lines down the screen.

Note that if you specify just a column number it is possible to display information from many records, one below the other starting at the same column, but if you specify both a column and a line number then only information from one record at a time can be displayed on the screen.

This is for the simple reason that two items of information can be in the same column by being one below the other, but cannot be in the same line AND column without one overwriting the other.

Two further forms of the command exist. The command @0 positions the cursor at line 1, column 1 of the current screen, without clearing the screen.

The command @x,0 will cause a carriage return to column x. Use @1,0 to force the printing of blank lines.

For the underlining command, see note 7 in Section 13.6.

Used in combination with the other output formatting commands, the position command gives you enormous flexibility as to how your output should appear on either screen or printer.

8.10 ABBREVIATING OUTPUT COMMANDS

Various examples were given above of output commands using the words 'the' and 'records' as in

```
'print down all the records [Ref]'
```

and

```
'display across the records from "hlist" [Ref]'
```

In fact these two words, and also the spaces between words, are not needed at all by Superbase and are allowed merely to make the syntax more natural.

These commands could instead have been entered as:

```
'printdownall[Ref]'
```

and

'displayacrossfrom"hlist"[Ref]'

We can even abbreviate further since all Superbase commands can be reduced to their shortest unique form, with a minimum of 2 letters. Usually you can type just the first letter followed by the second letter with the SHIFT key held down.

The previous two commands then become:

'prldOaL[Ref]'

and

'dIaCfR"hlist"[Ref]'

Once you are fully familiar with the commands you may wish to use these abbreviations to cut down the amount of typing required.

3.11 OUTPUT TO A FILE

Superbase is designed for integration with word processing packages, Easy Script in particular. Integration is achieved by outputting data (in CBM ASCII code) from a Superbase file into a file that can be read in by the word processor. The command is entered on the command line in the usual way when the OUTPUT Option is selected from Menu 1.

There are 2 forms of the command. The first one outputs to a file exactly as to the screen or printer, following all truncation commands. If the output is DOWN, no field names will be output. If the output is ACROSS, field contents will be output one after the other on the same line, up to the maximum specified with the right margin setting command, RMARG. The command is:

'down all to "addresses" [Name] [Address1] [Address2]'

This would produce a file called 'addresses' containing the name and 2 lines of address from all the records in a Superbase file. Each field will appear on a separate line and each line will end with a carriage return character. A blank line with just a carriage return on it will be output between records.

A Key List may also be used:

'across from "hlist" to "addresses" [Name] [Address1] [Address2]'

This would use the Key List "hlist" (usually obtained with a FIND operation) to produce a file called 'addresses' containing the name and 2 lines of address from the selected records in a Superbase file. All 3 fields will appear on the same line with a space between fields, and each line will end with a carriage return character. A blank line with just a

carriage return on it will be output between records.

The other form of the command is especially designed for Easy Script, and uses the additional command FILL:

```
'all fill to "addresses" [Name] [Address1] [Address2]'
```

The use of FILL specifies that output will automatically be DOWN; that trailing spaces will be truncated from fields (unless the &[field] format is used); and that numbers will be output with 1 leading space for the sign, 2 digits after the decimal point, and left justified, unless an '&x,y' formatting command specifies otherwise. Each line will end with a carriage return and a blank line with just a carriage return on it will be output between records. (The blank line will need to be 'trapped' in a variable block in a comment line in an Easy Script document.)

8.12 ENTERING OUTPUT COMMANDS IN A COMMAND LINE

As with all of the Superbase commands it is possible to bypass the Menu selection and type in your commands directly as a COMMAND LINE from either of the Main Menus.

All you have to do is to prefix the commands with the Primary Command 'Output' as in

```
'output all the records [date][item]'
```

You can also combine output commands with others in the same Command Line such as:

```
'find "hlist" where [outstanding] is ">500" : output from "hlist"
 [Name];[address]'
```

Further examples of using the Output command can be found in Tutorial Two and more details about Command Lines can be found in Section 3.

9 FILE

9.1 FILES AND DATABASES

Superbase organizes your stored information in files of records, each of which has a different filename and may have a different record format.

These files are held in groups of fifteen and each such group is known as a database. Information may be exchanged between databases by using the IMPORT and EXPORT commands, available from the MAINTAIN Menu. **Never remove a database disk and replace it with another without using the 'DATABASE' command.** The procedure is to return to either Main Menu, change disks, and type the command: '**database "name"**' where "name" is the name of the new database.

9.1.1 OBTAINING THE FILE OPTION

The FILE Option is used whenever you wish to change the file you are currently working with to another file in the database or to create a new file.

From Menu 2 press the key marked 'f1' on the right hand side of the keyboard.

The message 'Mode: Enter Filename' will be displayed in the message area at the top of the screen and the main screen area will display a list of files available in the current database. Superbase will wait for you to type in the name of the file you wish to select, which may be either one of the existing files listed on the screen or the name of a new file that you want to set up.

New files may be added to the current database up to a maximum of fifteen files.

If you have selected an existing file, you will be returned to Menu 1. Notice that the 'File Selected' Indicator shows the name of the file you have chosen.

You may now use any option from the Menu, such as ENTER if you want to enter information into the file, or SELECT if you want to access a particular record.

If, on the other hand, you have selected a file that does not exist, the screen will clear and the following prompt will appear:

'File Does Not Exist: Create It?'

If you respond 'y', you will be placed in the FORMAT Option, ready to design the screen layout for the new file. If you respond 'n', you will be returned to Menu 1. The new filename will not show in the 'File Selected' Indicator until the format has been completed.

Directions on how to FORMAT a new file are given in Section 4.

9.2 CHANGING FILE BY DIRECT OR PROGRAM COMMAND

A new file may be selected using the 'file' command followed by the filename in quotes.

Examples

```
'file "cust.rec"'
'a$="addresses":file a$'
```

The above commands avoid the standard 'Enter Filename' prompt and allow programs to transfer to another file without user intervention.

The command 'file' on its own in a program will cause the 'Enter Filename' prompt to appear on the command line and await operator response. See the start program at Appendix C.

9.3 SUMMARY

The FILE Option is used to access a file from the current database. This may be either an existing file or else may be a file that you want to set up.

Selects the File Option from Menu 2.

After typing a Filename, accesses that file if it already exists, or gives you the option to FORMAT the new file if it does not exist. The database must also have fewer than fifteen files created.

10 SORT

10.1 SORTING THE RECORDS

Superbase automatically keeps the records in a file sorted alphabetically according to the contents of the KEY Field of each record.

You may, however, want either all of the records or the records in some Key List sorted according to the contents of a field other than the KEY Field.

For instance, in a file of invoice records you might want to SORT the records according to the amount each customer owes you.

With the SORT command you can create a sorted list which is just like a Key List except for the order in which the records are listed. This list can then be used as a Key List in the OUTPUT, BATCH, REPORT and SELECT options.

10.1.1 OBTAINING THE SORT OPTION

From Menu 2 press the 'f4' key.

10.2 ENTERING THE SORT PARAMETERS

On entering the SORT option you will be prompted with:

'all/from "list" (item list....)

If you want to produce a sorted list of all the records in the file, type 'all'. Otherwise type 'from' followed by the name of the Key List of records to be sorted. The (item list....) signifies that field names and/or variables may be included in the command.

You can omit the name of the Key List and just type 'From""' in which case the current default Key List, 'hlist', will be used (see Section 7.2).

You can also choose a DESCENDING SORT or an ASCENDING SORT. A Descending Sort will sort the records in a reverse alphabetic or numeric order and is specified by typing 'D-' after the selection of records to be sorted. The order is from z to 'space', including from 9 to 0. If you leave out the 'D-' then Superbase will assume that you want a normal Ascending Sort.

Next you must specify which field or fields you want the sort to be performed 'on', enclosing the name of each field chosen in square brackets. If you specify more than one field, Superbase will take those records having the same contents in the first specified field and sort

those according to the second field specified and so on. Do not specify the key as a field to sort on, as it is automatically included at the end of the list.

The sort will use ten characters from each chosen field unless otherwise specified by the Field Truncator command '&' (see section 8.9.1). For instance,'on &6[Name]' will only use six characters from the Name field and ignore any others for the purposes of sorting.

To include the whole field contents in the sort, even if the length is greater than ten characters, '&[Field]' can be used. The maximum number of characters to sort on is 256 minus the length of the key.

Finally you can specify the name of the new list of records which the sort will produce by typing 'TO "new listname"'. This will result in the sorted list being stored on disk under that name. If you leave out the 'TO' command then the sorted list will be stored under the name "hlist" as in the FIND Option.

After typing in these parameters, such as:

'from "hlist" D- on [Outstanding] [Ref] to "debit-list"'

or perhaps

'all on [item]'

press RETURN and the message 'Processing' will be displayed at the top of the screen. During the sort Superbase may require space for its intermediate sort file called 'hslist'. This file, if created, will be 8K in size initially, and will increment in size by 8K each time it is recreated during the process. You can delete any 'hslist' files left on the disk after a sort if you need to create space. A record counter is used during the sort to indicate the number of records sorted and written to disk. After Superbase has finished creating the sorted list you will be returned to the Menu 1 and the sorted list will be ready for use.

Note that the fewer or shorter the parameters specified, the quicker the sorting process will be.

11 CALCULATION

11.1 THE CALC OPTION

The CALC Option is provided to enable you to carry out calculations which can be on the information in the fields of your current record.

As well as normal arithmetic operations, the whole range of BASIC functions including trigonometric functions and string functions is available.

The results of your calculations can be stored in the fields of the record, stored in BASIC variables or simply displayed on the screen.

RESULT Fields and CALENDAR Fields cannot be altered by the CALC Command. KEY Fields can be set and the new record can be added to the file. See Section 11.4.

11.1.1 OBTAINING THE CALC OPTION

From Menu 1, press the 'f5' key. The message 'Enter Calculation' will be displayed at the top of the screen.

11.2 ENTERING CALCULATIONS

The simplest type of calculation you can enter is a straightforward expression such as '0.15*48.60'. This would display the result of the expression, namely '7.29'. Any such expression entered which does not include an '=' sign will cause the result of the expression to be displayed in this way.

Expressions can of course be a good deal more complex, such as 'cos(sin(9)/log(10))↑2' and may include string functions such as 'left$([Name],3) to obtain just 3 characters of a field.

Using the '=' sign will cause whatever is to the left of the '=' sign to be assigned the value of whatever expression is to the right. For instance '[Price]=18.50' will assign the value 18.50 to the PRICE field in the current record. Similarly, '[Name]="Horace";[Address]="35 Sunnyview Crescent"' will result in the NAME and ADDRESS fields being changed accordingly. Note that when assigning strings of text to non-numeric fields, the text assigned must be put in double quotes as above.

It is important to remember, however, that when the contents of fields are modified in this way the modifications are only retained temporarily unless you issue the STORE command to tell Superbase that you want the

modified version of the record to be stored on disk.

In fact without the STORE command the modifications will only be retained until another record becomes the current record. This is useful since you may be modifying numeric fields just to see what effect the modification would have on the RESULT fields in the record. In this way, the CALC option provides you with a 'What if?' facility for exploring the effects of various changes such as changes of discount rates or tax rates or whatever.

The STORE command can be typed in when you have returned to the Main Menu after using the CALC option. Simply type 'Store' and press RETURN.

Also note that CALC lines involving more than one calculation must have the individual calculations separated by semicolons.

11.2.1 STORING RESULTS IN BASIC VARIABLES

Another possibility is to assign values to BASIC variables such as 'x=[Price]-[Price]*0.15', which would give 'x' the value of the price minus a fifteen percent discount. 'x' could then be used in further calculations either in the same CALC Command or later on in a new calculation.

This facility can be exploited in the BATCH option to produce running totals (see Section 12.3).

Since any expression not including the '=' sign displays the result on screen, you can evaluate an expression and then display the result as in 'x=[Price]-([Price]*0.15);x'. Information to be displayed in this way should always be the last item in the CALC line. Only BASIC variables, not fields can be displayed in this way.

The examples above are both examples of assigning numeric results to variables. It is also possible to assign strings of TEXT to variables, but care must be taken that TEXT VARIABLES are used for this purpose as in:

```
'x$="1 Portable TV"'
```

A text variable differs from a NUMERIC VARIABLE in that it always ends with a '$' sign as in 'a$', 'a1$' and 'x$'.

11.3 ENTERING CALCULATIONS AS COMMAND LINES

As with other Superbase commands, you can type a CALC command in directly from either Main Menu (see Section 3). Just type in a CALC line as above but with the Primary Command 'CALC' preceding it as in:

```
'Calc x=[Price]/2;y=[Price]/3;x;y'
```

which would display one half and one third of the Price field in the current record.

Calculations can be combined with other Superbase commands as in

```
'select match where [Name] is "=Robins":calc x=[Due]*[Interest];
[Due]=[Due]+x;x:store
```

Note the use of the STORE command here to make the modification to Robins' record permanent.

Remember to separate the various Command Clauses by colons and the various calculations within the CALC Command Clause by semicolons.

If you modify a field with CALC and wish to display it subsequently, you must use a further CALC clause to do so. This is because all references to a field within the same clause will use the original value ('value' here meaning both text and number value) of the field. However, you can assign the value of a modified field to a BASIC variable and display that all in one command.

11.4 ENTERING A NEW RECORD WITH THE CALC COMMAND

Since CALC can be used to set the contents of fields, it can be used to enter a new record into the file. To do this however, you must first CLEAR the current record by issuing the CLEAR command and you must end by storing the new record with the STORE command. It is also mandatory to enter a KEY in the KEY Field. You cannot CALC a RESULT or a CALENDAR field, only the fields that they are derived from. Nor can you CALC a replica field. You can, however, CALC a CONSTANT field.

The CLEAR Command which is detailed in the Programming Section with the STORE command and others, will not delete the current record but will instead make it cease to be the current record. The new current record will be an empty record into which you can then enter information into by setting the fields equal to some expression in a CALC command.

An example would be:

```
'clear:calc [No.]="01670";[Item]="1 FR80 Camera";[Price]=79.85:store'
```

where [No.] is the KEY Field.

If there are any errors in the CALC commands, the STORE command will not work, so you cannot create invalid records.

12 BATCH

12.1 THE BATCH OPTION

The BATCH option, like the CALC option, is for carrying out calculations on the information in your files. But whereas the CALC option performs calculations on the fields in the CURRENT record, the BATCH option carries out calculations on the fields in either ALL of records in a file or on a predefined selection of records using a KEY LIST (for details about setting up a Key List see Section 7).

This ability to perform operations on selected fields of records throughout the file makes UPDATING your files a simple, automatic process.

12.1.1 OBTAINING THE BATCH OPTION

To obtain the BATCH option from Menu 2, press the 'f3' key.

12.2 PERFORMING CALCULATIONS ON ALL RECORDS OR SELECTED RECORDS

When you enter the BATCH option you will be prompted with:

'Enter: all/from "list" (item list....)'

If you type 'All' before the calculations you specify, they will be performed throughout the file. If you type 'From "listname"' then they will only be performed on the records specified in the Key List you have named. The '(item list....)' signifies that field-names and/or variables may be included in the command.

The calculations themselves are entered in exactly the same way as in the CALC option detailed in the last section. You do not have to use the STORE command to make modifications to fields in the records permanent. The BATCH command automatically stores the modified records for you.

An example of using BATCH to update a file such as the Customer Invoice file from Tutorials Two and Three is as follows:

```
'from "Pending-list" [Total-paid]=[Total-paid]+[Last-paid];
[Pending-Update]="No"'
```

12.3 CALCULATING TOTALS

The BATCH option can also be used to display running totals of fields in the records. Suppose you wanted to total the amount outstanding on all invoices in the Customer Invoice File. The BATCH command for this would be:

```
'all x=x+[Outstanding];x'
```

Here the BASIC variable, 'x', is being used as an ACCUMULATING variable. SUPERBASE will go through all the records in the file, adding the value in the OUTSTANDING field to the previous value of 'x' and displaying the result. In this way 'x' will accumulate the values in the OUTSTANDING field for each record. Note that the variable should be set to a value of zero before you begin.

12.4 USING THE BATCH COMMAND IN A COMMAND LINE

Like all of the other Primary Commands, BATCH commands can be entered directly from either of the Main Menus. Simply prefix the command with the word 'Batch' as in:

```
'batch all x=x+[Outstanding];x'
```

The BATCH command can also be combined with other commands to form a more complex Command Line as in:

```
'find "hlist" where [Item] is "Radio";[Quantity] is ">10":
batch from "hlist" [Discount]="12%"'
```

This Command Line would find all the invoices for more than ten radios and adjust the discount rate on those invoices to twelve percent.

13 REPORT

13.1 THE REPORT GENERATOR

The REPORT Option provides you with a REPORT GENERATOR for producing printed reports based on information from your files including TOTALS and SUBTOTALS calculated on any field.

The reports can be formatted in a number of different ways with the commands used in the OUTPUT Option (see Section 8). Text of your choice and calculations of the kind available in the CALC and BATCH Options can be included.

The REPORT GENERATOR is entirely prompt driven. In other words you are given a series of prompts, the responses to which determine the nature of the report produced.

From your responses to these prompts the Report Generator will produce a PROGRAM which creates the report itself. This program becomes the current program in memory and can be modified like the user-created programs of the PROG option or can be simply executed or stored on disk for future use.

The report can either be displayed on screen or printed out on paper depending on which of the commands 'Display' or 'Print' was last used. You can change from one to the other by typing either 'Display' or 'Print' from either Main Menu before executing the report.

13.1.1 OBTAINING THE REPORT OPTION

From Menu 1 use the 'f6' key to obtain the REPORT Option. The 'f6' key is the key marked 'f5' on top and 'f6' on the front, pressed while the SHIFT key is held down.

13.2 USING THE REPORT GENERATOR

Throughout the REPORT Option you will be given a series of prompts regarding the parameters of the report to be produced.

SELECTING THE FILE

On entering the REPORT Option you will be asked:

'Enter File to Report on'

Type in the name of the File containing the information you wish to include in the report and press RETURN. Note that the file name must be enclosed in quotation marks.

All parameters from here on are optional (with the exception of the 'All or From "list"' prompt, which must have one or the other specified). If you wish to omit any of them, just press RETURN when you receive the prompt concerned.

ENTERING THE TITLE

The next prompt is: .

'Enter Report Title'

The Report Title, which must also be enclosed in quotation marks, can be as long as you like and can be positioned on the page with the POSITION command '@' (see Section 8.9.2).

Your title may be just a line or two of text to head or to explain the nature of the report, or it may include the headings of any columns of information in the report. To force a new line, use the formatting command '@1,0'.

If your title takes up all of the 79 characters available in the command area at the top of the screen you should split the title up and enter it in parts. You will be prompted with:

'Any More?'

If you want a longer title than you have so far entered then type 'Y' for 'Yes' otherwise type 'N' for 'No'.

TOTALS AND SUBTOTALS

After entering your title and pressing RETURN you will be prompted with:

'Enter Total Calculation(s)'

Each report can contain up to ten TOTALS on any of the fields in the file you have specified. These totals, which can be printed at the end of the report are designated t0, t1,..., t9 and can be specified by using any of the parameters available in the CALC option. A simple example which would total the amount due from every record is 't1=t1+[due]'. Multiple totals may be specified, and must be separated with semicolons.

You could also specify calculations on the totals such as:

't1=t1+[price]*0.8; t2=t2+[discount]-[discount]*0.1'

Another option open to you at this point is to specify up to ten SUBTOTALS which are printed at various break points in the report. The subtotals are designated by s0, s1,..., s9 and are specified in the same way as totals:

's1=s1+[quantity]

Again, any of the calculations available in the CALC Option can be performed on subtotals.

If you want to specify more totals and subtotals after using up the available space, respond 'Y' to the 'Any More?' prompt.

SUBTOTAL BREAK

After specifying the totals and subtotals you will be prompted with:

'Enter Field for Subtotal Break '

Subtotals will be printed every time the field you specify as the field for subtotal break changes its value.

For example, you may be producing a report about sales of various goods based on a file containing quantities of goods sold. Suppose you want a subtotal of the number of each product sold. You would specify a subtotal such as 's0=s0+[quantity]' and respond to the Subtotal Break prompt with '[goods]'.

USING A LIST OF SORTED RECORDS

SUBTOTAL BREAK FIELDS are only relevant if you produce a key list sorted on this field using the SORT command, before executing the report. If you do not do this first then the records will be printed in the ordinary alphabetical order of their key fields, and the value of the field specified as the Subtotal Break Field will change arbitrarily, giving you subtotals with no significance.

Having sorted the records you should use the name of the SORT LIST when asked 'All or From "list"' (see below).

If you want more than one Subtotal Break Field then specify all of the break fields in the SORT.

THE SUBTOTAL TEXT

Next, a prompt of

'Enter Subtotal Text'

will allow you to specify any text to be printed when the value of the field for subtotal break changes. The text must be inside double quotation marks. You might wish to print the words "Subtotals for department" each time the 'department' in a Sales Report changes. The quoted text may be preceded or followed by the subtotal variables to be printed, i.e. s0, s1, etc. Only if these are specified in the first line of Subtotal Text will they be automatically set to zero after being printed. Type in the text and commands you require and press RETURN.

You will again be prompted with:

'Any More?'

If there is to be more than one Subtotal Break Field then type 'Y' for 'Yes' otherwise type 'N' for 'No'.

USING A KEY LIST

The next prompt displayed is:

'Enter All or From "list"'

As in other options like OUTPUT and BATCH you can specify whether you want ALL of the records in the file to be used or just those in a particular KEY LIST. Type 'all' or 'from' followed by the name of the key list enclosed in quotation marks and press RETURN.

SELECTING THE FILE

The next prompt governs the main details in the report. You are requested to:

'Enter Report Detail'

The detail may consist of the contents of fields, strings of characters or the values of BASIC variables. You can use computed fields such as '[Sales] - [Costs]'. Most often the Detail section will consist of fields with text to describe what they are such as:

'"Goods: ";@8[goods];"Price: ";[price]'

If you are producing a report consisting of columns of information, however, it would be preferable to use column headings in the TITLE section and just use the word 'TOTAL' in the ENDREPORT section.

Any of the facilities provided with the OUTPUT Option can be used to specify and format the report detail (see section 8) and there are additional commands available such as LMARG, RMARG, PLEN, TLEN, LFEED, SPACE, CONT, PDEV, PDEF and PLUS which are detailed in the Programming Section.

END OF REPORT

The next prompt in the report sequence is:

'Enter End of Report Text'

You should enter here any line of text and totals you want printed at the end of the report such as '"Total Due is";t1'.

SAVING THE REPORT PROGRAM

Having specified the report parameters in this way you will next see the Report Program itself displayed on the screen and will be asked if you

want to save the Report Program on disk for future use. The prompt is:

'Save Report Definition?'

Type 'Y' for 'Yes' if you do want to save it otherwise type 'N' for 'No'. If you type 'N', the report program will still be available for execution as the current program in memory and you will be returned to Menu 1.

If you answered 'Y' to the last prompt then the final prompt in the report sequence is:

Enter Report Name'

Below the prompt you will see 'Save "' followed by the cursor. Type the name you want to give the Report Program, prefixing it with '1:' if you want to store it on drive 1 of a dual drive unit, and press RETURN. The Report Program will be stored on disk and you will be returned to Menu 1.

13.3 EXECUTING A REPORT PROGRAM

Once you have created a Report Program you can produce the report itself by using the EXECUTE Option (see Section 14).

Either select the EXECUTE Option by pressing the 'f7' key from Menu 1 or type the direct command 'Execute' from either Main Menu.

13.4 LOADING A REPORT PROGRAM

By storing Report Programs on disk you can build up a library of different types of Reports for future use. If you want to produce a Report from one of these Report Programs and it is not the CURRENT PROGRAM then you must first LOAD that Report Program by issuing the LOAD command from either Main Menu.

Just type 'Load "reportname"' where 'reportname' is the name you gave to the Report Program when it was created. That Report Program will then become the Current Program and can be executed in the normal way.

13.5 OTHER WAYS OF CREATING A REPORT PROGRAM

Once you have become thoroughly familiar with the procedure of setting up the Report Program there is no reason why you shouldn't produce reports by modifying already existing programs or even writing your own. Details about creating and modifying programs are given in the Programming Section.

13.6 ADDITIONAL REPORT PROGRAM FEATURES

To complete your knowledge, or if you are going to edit the programs produced by the Report Generator, you should be aware of the points discussed in this section.

1 Use the 'plus' command to obtain multiple lines of title, totals, subtotal text, detail, and report ending. See Programming Section 2.1.31.

2 Subtotals are only cleared to zero if they are declared in the first line of a subtotal command. Subtotal variables declared in this way are printed before being cleared to zero if they appear in subsequent program lines. However, there may be applications that need a large amount of text to be printed, requiring the use of plus commands to create multiple subtotal text lines. In these cases, if the subtotals to be printed appear on lines other than the first, they would not be cleared. To overcome this, type the subtotals to be cleared after the plus command but on the first line of the subtotal command. See point 3 for an example.

3 To retain the old subtotal break field for printing after a change of subtotal break field has been detected, assign the subtotal break field to a string variable as part of the total line: e.g. 'a$=[town]'.

Here is an example of a subtotal line that uses this feature:

```
subtotal [town] plus s1 s2
@5,0 "SUBTOTALS FOR " a$ " are" @40 &5,2 s1 @50 &8,2 s2 plus
@41"--------------------"
```

This example also makes use of the feature described in point 2.

4 If you omit both the 'all' and the 'from' from your detail line, the report will be from the current record. You can cause the current record to be selected according to your own requirements by appropriate programming (with select match, for example).

5 The report and endreport commands are compulsory in a report program.

6 If you have difficulty printing the £ sign, you can use a direct command to assign the necessary escape sequence (which should be described in your printer manual) to a BASIC variable before printing the report. For example, 'P$=chr$(27)+"Y"' will work for some printers. Then when 'P$' is placed in the report detail line, a £ sign will appear. The same technique can be used to access other characters such as other currency signs.

7 You may wish to produce underlined text or fields if your printer supports this feature. Two forms of the '@' command are provided:

```
@-[Field]@-
```

This switches underlining on for [Field] and then switches it off again.

@+[Field1] [Field2]

This switches underlining on for the next text item only. In this case, 'Field1' will be underlined, but not 'Field2'.

If this command is used with screen output, the underlined areas will appear in reverse video. On Commodore matrix printers (which do not support underlining) the reverse effect will also appear.

8 Single-column labels may be printed without programming. Multi-column labels may be printed with a program that stores details of the name of more than one record in an array in memory, then prints them out line by line. A utility program for printing labels is provided on the Superbase disk. Load "labels" from the Superbase disk and save it onto your own data disk. The program may be edited for your own special needs.

See the HELP topic "labels" for detailed information.

14 EXECUTE

14.1 THE EXECUTE COMMAND

The purpose of the EXECUTE command is to run a Program previously created either by means of the PROG Option or by numbering individual Command Lines (see Section 15).

Programs are lists of numbered COMMAND LINES made up from the Superbase commands, both those appearing on the various Superbase MENUS and the extra commands detailed in the Programming Section. You can also use any BASIC commands in a Program. This enables you to automate sequences of Superbase operations.

14.1.1 OBTAINING THE EXECUTE OPTION

To obtain the EXECUTE option press the 'f7' key from the Main Menu.

14.2 EXECUTING THE CURRENT PROGRAM

The Program in memory is called the current Program. There are two ways in which Superbase holds Programs: on disk and in memory. The last Program created will always be the one in memory.

If it exists, this is the Program that will be executed when you select the EXECUTE option from the Main Menu. If there is no Program in memory, the message 'Enter Program Name' will be displayed.

14.3 EXECUTING PROGRAMS FROM DISK

Programs can be given names and stored on disk. They can be executed directly by typing 'Load "progname":Execute ' from either Main Menu.

Alternatively they can be executed as follows:

Remove any existing Program from memory with the 'NEW' command. See below.

Select the EXECUTE command from the Main Menu.

Enter the name of the Program required when prompted with 'Enter Program Filename'.

14.3.1 REMOVING THE CURRENT PROGRAM FROM MEMORY

Since the current Program is automatically executed when you select EXECUTE from the Main Menu, you must first remove it from memory if you want to execute another Program in this way.

The Command NEW is provided for this purpose and can be typed in from either Main Menu. The NEW command could also be included as the last executed command in the Programs themselves so that once a Program has been executed it will be removed from memory.

14.3.2 SAVING PROGRAMS ON DISK

If you want to save the current Program on disk for future use you should use the SAVE command. Programs may be protected on disk with the PROTECT command.

> **Note that once protected, Programs cannot be loaded and edited, so ensure that you take an unprotected copy first.**

From either Main Menu type 'Save "progname"' and press RETURN. You can give a '1:' prefix to save on drive 1 of a dual drive unit. The Program will be saved permanently on disk under the name you have given it.

You may wish to build a library of Programs on a separate disk. If you exchange disks (while in the Main Menu) in order to load a Program, you should re-initialize the database from the Main Menu with the 'database' command.

14.3.3 LOADING PROGRAMS

Once you have Programs stored on disk you can call up any of them to take the place of the Program currently in memory. Use the DIRECTORY Option from the Maintain Menu to look at your programs. They are the ones with a '.p' suffix, such as 'start.p'.

From either Main Menu type 'Load "progname"' and press RETURN.

For more information about Programs and the NEW, LOAD and SAVE commands see the Programming Section.

15 MAINTAIN

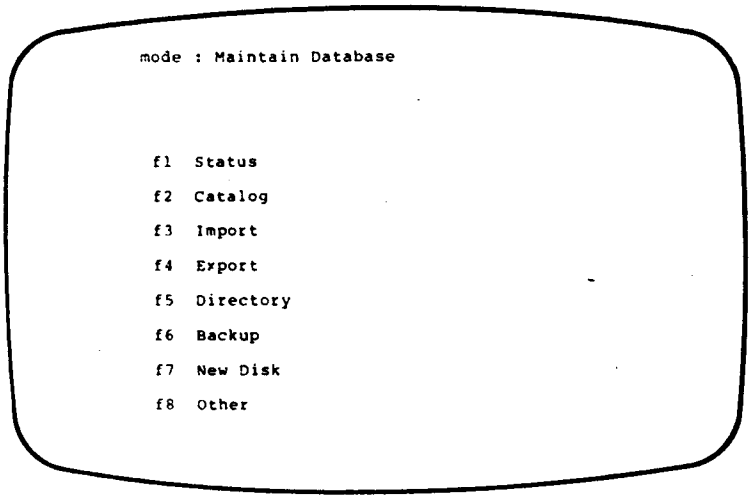
15.1 THE MAINTAIN OPTION

The Maintain Option provides a series of utilities for performing various operations on your databases. It allows you to obtain either a list of the fields in your current file together with their types and lengths, or a list of all the files in the current database. It also enables you to transfer data to and from an external program, and to carry out various disk commands including a single-drive back-up, a disk formatting command and a disk directory command.

15.1.1 OBTAINING THE MAINTAIN OPTION

From Menu 2 press the f6 key.

You should see a further menu screen. This is the Maintain Menu.



THE MAINTAIN MENU

The Maintain Menu offers eight sub-options each of which is detailed below alongside the function keys required to obtain them. Certain items are covered more fully later in this section.

STATUS

This option provides a list on the screen of each of the fields of the current file along with their field-types and their lengths. The output may be directed to the printer.

CATALOG

A list of all of the files in the current database can be obtained with this option. This database CATALOG is the same as that provided by the FILE Option (see Section 9) when you want to change the file you are working on. The output may be directed to the printer.

IMPORT

This option allows data from external programs to be read into Superbase. Deselect with RETURN.

This may be data from another Database Management System you wish to convert to Superbase or it may be data from a different type of program altogether. You can also re-import data that has been exported by Superbase as a means of transferring files from one of Superbase's databases to another.

For full details see Section 15.3.

EXPORT

The EXPORT Command is used to produce a SEQUENTIAL File of data from any Superbase file which can be kept as an ARCHIVE copy to be stored on disk or tape, or can be IMPORTED back into Superbase as a means of transferring data from one Superbase database to another. Deselect with RETURN.

These sequential files can also be used by external programs.

To output data to EASY SCRIPT see the command OUTPUT TO. Reference Section 8.11.

For full details see Section 15.2.

DIRECTORY

A list of all programs and files on a disk can be generated by this option.

SHIFT for f2 f4 f6 f8

BACKUP

This option allows you to make a back-up copy of your data disks. For full details see Section 15.4. Deselect by entering 'n' to the 'Are You Sure?' prompt.

NEW DISK

Formatting a blank disk or a disk you wish to re-use can be done by means of this option. Deselect by entering 'n' to the 'Are You Sure?' prompt.

In this way you can avoid having to leave Superbase to type in the Commodore disk formatting command.

OTHER

Other Commodore disk commands can be executed by means of this option. Deselect with RETURN.

Examples are renaming a file, copying a file and deleting files from the disk. For full details see Section 15.6.

15.2 IMPORT

The IMPORT command is the converse of the EXPORT command in that it allows you to read data into a Superbase file from an external SEQUENTIAL FILE.

One advantage of this is that if you already have a database system containing a large amount of data you can transfer the data to Superbase. To do this, set up a Superbase file with the same format as the one to be imported. Ensure that the selected Superbase file has at least as many fields as the file being imported. The NAMES of the fields do not matter but the order of the FIELD TYPES must be the same (e.g. TEXT or NUMERIC) in the file to be imported and the Superbase file set up to receive the data.

Note that the external file must first have been copied onto the Superbase DATA DISK unless you are using a dual-drive system. Files to be imported must be in Commodore ASCII format. In some cases, a conversion must be performed prior to the IMPORT operation. For example, certain screen code files must be processed to remove double quotation marks, strip trailing spaces, and change the end of line marker, chr$(31) to a carriage return, chr$(13). Public domain software exists for this purpose or the SUPERSCRIPT program on 2/3/4/8000 series Commodore computers may be used to create converted ASCII format files.

On selecting IMPORT from the Maintain Menu you will be prompted with 'Enter Import Filename'. If you have a single drive and are importing a sequential file in which the fields of data are just separated by 'RETURN's (see section 15.2), type just 'filename' and press RETURN.

If you have a dual-drive system you can type '1:filename' to specify that the source file is on a disk in drive one. If the source file has separators between its fields and records other than 'RETURN' then you

can also specify these from the Main Menu as in 'import"filename", "/."' which specifies slashes between fields and full stops between records.

The IMPORT command can also be used to re-import data from a sequential file created by EXPORT of data from another Superbase database.

Use IMPORT to transfer files across databases within Superbase itself.

When transferring files from 4040 to 1541 format, use the BACKUP Option to read a 4040 disk and copy it to a 1541 disk. Use the same technique when transferring from 1541 to 4040.

> **Note that the disk error light (and tone if fitted) will signal extensively during the IMPORT operation. This does not normally signify an error.**

15.3 EXPORT

The EXPORT command takes the data in one of your files and converts it into a SEQUENTIAL FILE. A sequential file is a sort of common currency in which information can be accepted by another program.

If the external program you wish to export data to uses something other than RETURN to separate the fields you can specify one or two separators after the file name.

On selection of the EXPORT option from the MAINTAIN menu you are prompted with 'Enter Filename'. If you respond with just a name as in 'Exportfile' the sequential file created will consist of all of the fields from your current file, record by record, with each separated by a 'RETURN'.

From the Main Menus you may type 'export"filename","/."' then the fields would be separated by slashes and the records by full stops (an ASCII code such as chr$(10), not inside quotation marks, may be used if you prefer).

The sequential file created by the EXPORT command can also be re-imported into another of Superbase's databases. For this purpose there is no need to specify special field and record separators. Use the FORMAT option to set up a record layout to accept the incoming data.

Another use for a sequential file that has been exported from one of your files is as a long term ARCHIVE for security storage on disk or tape.

Note, however, that the EXPORT command will export ALL of the fields from ALL of the records. If you want a field and record selective output then you should use the 'OUTPUT TO' command which is detailed in Section 8.

15.4 BACKUP

The BACKUP Option allows you to make duplicate copies of a disk. All files on a disk are copied. BACKUP handles both single and dual disk drives.

Owners of a CBM 2031 disk drive may use the single drive backup procedure provided a suitable interface is installed. The procedure will also work with drive 0 of a 4040 dual drive unit, but not with 8050 or 8250 units, which must use the dual drive backup procedure.

Owners of a hard disk unit should use the EXPORT Option to create a backup data file on the hard disk, which may then be copied to floppy disk or tape using DOS utilities. Superbase file formats, text files, programs, and lists may be copied freely. A Superbase program may be written to carry out hard disk backup operations.

Using BACKUP requires the full computer memory and consequently Superbase will close down when the process is complete. Reload as normal if you wish to continue.

The disk to be copied is referred to as the SOURCE DISK, and the disk onto which the data is to be copied is referred to as the DESTINATION DISK.

When BACKUP is selected the first question asked is:

Single Drive? (y or n)

Respond 'y' or 'n' as appropriate.

15.4.1 DUAL DRIVE BACKUP

The first screen prompt is:

Insert Blank Disk in Drive 1

Press Return to Continue

Insert the destination disk in drive 1 and press RETURN. The next prompt is:

All Data on Drive 1 Will Be Destroyed

Are You Sure?

This allows you to double-check that the disk you are using as the destination disk is the correct disk. It can be either a new disk from the box or an old disk that you want to re-use.

When you are satisfied that the destination disk is correctly inserted, press RETURN. Superbase will duplicate the disk on drive 0 onto the disk in drive 1.

15.4.2 SINGLE DRIVE BACKUP

The first screen prompt is:

Insert Blank Disk in Drive 0

Press Return to Continue

Insert the destination disk in drive 0 and press RETURN. The next prompt is:

All Data on Drive 0 Will Be Destroyed

Are You Sure?

This allows you to double-check that the disk you are using as the destination disk is the correct disk. It can be either a new disk from the box or an old disk that you want to re-use. When you are satisfied that the destination disk is correctly inserted, press RETURN.

PREPARE DISK

The next prompt is:

Enter Disk Name, id

You must give your disk a name and identification code. The name can be any combination of up to 16 characters, including spaces but excluding the colon (:). A meaningful name such as 'work disk' is advised. The identification code must be 2 characters, such as 'aa'. Give every disk a different code,'ab','ac', etc.

Type the disk name and code, including a comma between them, and press RETURN. The message 'Processing' will be displayed while the disk is prepared for use.

INSERT SOURCE DISK

When the disk is ready, the flashing message 'Insert Source Disk' will appear. Remove the newly prepared disk, insert the source disk into the drive, and press RETURN.

The message 'Please Wait' appears with a line of dots following it. The line extends dot by dot as Superbase reads in the data to be copied.

INSERT DESTINATION DISK

When Superbase is ready, the flashing message 'Insert Destination Disk' will appear on the screen. Remove the source disk and insert the destination disk. Press RETURN. Again the message 'Please Wait' appears with a line of dots showing that Superbase is

writing data onto the destination disk.

REPEATING THE CYCLE

If there is a lot of data to be copied, Superbase may require the process to be repeated, up to a maximum of 4 cycles. The message 'Insert Source Disk' will appear again if the cycle must be repeated. In this case, go through the actions as before, starting at the INSERT SOURCE DISK paragraph.

15.4.3 CONCLUDING BACKUP

When all data has been copied, Superbase will close down. resetting the computer to the state it was in just after it was switched on. To continue using Superbase, reload it in the normal way.

15.5 NEW DISK

If you require a formatted disk you may use this option to create one. A disk formatted with this process will be suitable for storing a database and the files, lists, and programs that go with it. However, the supplied HELP screens and the 'start' program will not be present. If you require a disk to have these on it you should create a new disk with the facility available when Superbase is started up.

The process is like the first stage of the BACKUP process.

The first screen prompt is:

Insert Blank Disk in Drive 0

Press Return to Continue

Insert the destination disk in drive 0 and press RETURN. The next prompt is:

All Data on Drive 0 Will Be Destroyed

Are You Sure?

This allows you to double-check that the disk you are using as the destination disk is the correct disk. It can be either a new disk from the box or an old disk that you want to re-use. When you are satisfied that the destination disk is correctly inserted, press RETURN.

The next prompt is:

Enter Disk Name, id

You must give your disk a name and identification code. The name can be any combination of up to 16 characters, including spaces but excluding the colon (:). A meaningful name such as 'work disk' is advised. The identification code must be 2 characters, such as 'aa'. Give every disk a different code,'ab','ac', etc.

Type the disk name and code, including a comma between them, and press RETURN. The message 'Processing' will be displayed while the disk is prepared for use.

During the process the current database is deselected. To restart type 'database' and press RETURN. Then enter the database name, without quotation marks.

15.6 USING THE DISK COMMANDS

The 'OTHER' option allows you to type in certain Commodore disk commands without exiting from Superbase.

You may use the following disk commands:

Rename, Copy and Scratch - described below.
vn: Validate
$n View disk directory (pattern matching can be used)

For full information on these and other disk commands see the manual which comes with your Commodore disk drive.

Note that if you are using the 1541 single drive unit you should type '0' (in place of 'n') for the drive number.

The following Disk commands may not be used within Superbase:

in:	Initialize disk	
un:	Reset	
nn:	New	(See Section 15.5)
dn:	Duplicate	(See Section 15.4.1)

RENAME

The RENAME Command renames an existing file. The new name specified in the command must not already exist. If it does, 'FILE EXISTS' error message will be displayed. The format of RENAME is:

'r0:newname=0:oldname'

If you rename a database, it must be in UPPER CASE letters. Do not rename a data file inside the database. The database would not be able to identify the records belonging to that file. Be sure to rename Programs with a '.p' suffix, and HELP screens with 'h' as the first letter. These rules also apply to any renaming done with the COPY command.

COPY

The COPY Command allows you to create multiple copies of files (under different names if they are to be on the same disk). Do not attempt to make a copy of a database: the only way to do this is with the EXPORT and IMPORT commands. The format of COPY is:

```
'c0:newname=0:oldname'
```

SCRATCH

The SCRATCH command is for removing unwanted files from the disk. No UPPER CASE filename can be scratched. The format of SCRATCH is:

```
's0:filename'
```

You will be asked 'Are You Sure?'.

To delete a data file from the database you should SELECT DELETE all the records in the file and then do an extra SELECT DELETE to delete the file. Then delete the file definition using the SCRATCH command.

A database may not be scratched from within Superbase. If you really have to remove a database, use the following BASIC program:

```
10 open 1,8,15,"s0:databasename"
20 input#1,er,em$,et,es
30 print er;em$;et;es
40 close 1
```

This will scratch all indexes and data within the database. Programs, screen definitions and other files are unaffected.

Use this routine to remove the TRAINING database from your disk and create an uncluttered source disk.

16 MEMO

16.1 USING THE MEMO OPTION

The MEMO Option has a dual purpose. It can be used to write and store messages to oneself or to other operators of Superbase which can later be recalled and read, and it can also be used to produce or amend a HELP SCREEN to provide advice to users having difficulties with any of the options of Superbase (see Section 17.1).

16.1.1 OBTAINING THE MEMO OPTION

From Menu 2, press the 'f7' key on the right hand side of the key board.

You should see a blank screen with the words: 'Enter Name' in the message area at the top of the screen.

You may call your MEMO by any name you choose up to 16 characters long, but if you want it to be a HELP screen, recallable by means of the HELP option, then you must prefix the name with an 'h'.

For example, if you want to create a HELP screen which offers advice on the IMPORT facility of the MAINTAIN option (see section 15.5), you should call your MEMO 'himport'.

Type in the name you have chosen and press RETURN.

If a MEMO already exists with the name you have typed, that MEMO will now appear on the screen for you to read or to edit with the flashing cursor at the top left.

The words 'Mode: Memo Writer' will be displayed in the message area above the screen.

If there is no MEMO with that name, the screen will remain blank and the cursor will appear ready for you to create it.

The following control keys are available for use while creating your MEMO screen:

These keys are used to move the CURSOR to the LEFT or RIGHT across the screen.

These keys are used to move the CURSOR UP or DOWN the screen.

This key takes the CURSOR directly to the 'HOME' position at the top left of the screen.

Characters can be INSERTED or DELETED with this key.

REFERENCE - MEMO

If you wish to **DELETE** a line from the memo screen use this key. All subsequent text will be moved up to fill the gap left by the deleted line.

You can also **ERASE** a line without moving up the text which follows it. Use this key to erase the line and whatever text was on that line will be replaced by blank spaces.

Use this command to print the current MEMO screen.

Lines can be **INSERTED** into the Memo screen with this key. Subsequent lines will be moved down to make room for the new line that has been inserted.

If you want to **CLEAR** the whole screen to start again, use this key.

You can **QUIT** the Memo option and return to the Main Menu by using this key.

Any changes you have made to an existing memo or any new memo you have created will be disregarded.

This option is mainly for returning to the Main Menu if you have entered the Memo option to read an existing Memo.

When you have completed the Memo screen, use this key to **STORE** the memo.

The word 'Finished' will appear in the message area and after a short delay you will be returned to the Main Menu.

Memos may be multi-screen. Create each page separately with a different name. Use the disk copy command (see Section 15.6) to join the files together: c0:newfile=0:memo1,memo2,memo3, up to 4 joined files. The large file may be edited by moving forward from page to page with the normal finishing command.

Note f1 Q must be used with caution as it erases the current and all subsequent screens.

17 HELP

17.1 HELP SCREENS

The HELP Option enables you to display special screens for advice on the use of Superbase facilities.

Superbase has a Help screen already provided for each of its major options, but you may design your own by using the MEMO Option (see section 16.1).

The first line of the built in HELP SCREENS follows a standard pattern. On the left a reference is given to the User Manual for further details. On the right the name of the Help Screen is given in capitals followed by the page number you are viewing.

You may obtain a printed copy of any Help Screen by holding down the CONTROL key and pressing 'P'.

17.1.1 OBTAINING THE HELP OPTION

From either Menu 1 or 2, press the 'f8' key.

You should see the message: 'Enter which help required' in the message area at the top of the screen.

Type in the name of the option which you want advice about, e.g. 'Format', and press 'Return'.

You will see the Help Screen displayed line by line and then the message 'End of Help, Press Any Key'.

When you are ready, press any key to return to the Main Menu.

1 PROG

1.1 THE PROG OPTION

The PROG Option is a special program editor with which users can write their own Superbase programs. PROG is the means by which Superbase becomes not only a Database Management System but also a powerful Applications Generator and Database Programming Language.

Many users will confine themselves to using just the Menu-driven system of control which is perfectly adequate for routine Database Management. The PROG option however, enables more advanced users to write their own PROGRAMS utilizing more than forty additional commands that Superbase provides in addition to BASIC.

These user-written programs can be designed to automate complex sequences of Superbase operations which can then be executed with just one key, the EXECUTE key, 'f7'.

It is even possible to use the PROG Option to set up user-defined Menus to give access to these programs in the same way that the Main Menus give access to the built-in options of Superbase.

1.1.1 OBTAINING THE PROG OPTION

To obtain the PROG Option, press the 'f5' key from Menu 2. The message 'Program Writer' will appear above an editing screen.

1.2 USING THE PROG OPTION

An area of the computer's memory is reserved for users' programs. Any program in this area is called the "current program". An existing program may only be edited after it has been loaded into the program area.

An existing program that is to be edited must be loaded from the main menus, before the PROG Option is selected. To load a program, type 'load "programname"' and press RETURN (omit the .p extension). When the program is loaded, select the PROG Option. If there is no current program (i.e. after a 'new' command), PROG will display a blank screen. You should use the 'new' command from the menu before starting to write a new program.

The program editor presents an editing window of 10 double lines (maximum length 79 characters) of program text. This may be scrolled up and down with cursor control keys. Programs are written by entering line numbers and Superbase commands in the desired sequence. A knowledge of the BASIC programming language will help the user to construct effective programs.

PROGRAMMING - PROG

The editing facilities of PROG are described in the next section. When a program has been written, exit from the editor by pressing f1 followed by RUN/STOP. The program will be held in memory, and may be executed by pressing the f7 'Execute' key.

To save a program after exiting from PROG (i.e. while at the main menu), use the command 'save "programname"' (omit the .p extension - this is supplied automatically). Specify a "1:" prefix if you wish to store the program on drive 1 of a dual drive unit.

1.2.1 EDITING FACILITIES

Vertical scrolling	Use the cursor up and down arrows key.
Horizontal movement	Use the cursor left/right arrows key.
Insert and delete	Use the INST/DEL key.
Abbreviate commands	Optionally use short form commands which will be expanded when the program is listed.
Redisplay screen	Use SHIFT/RETURN to redisplay after changing order of line numbers, etc.
End of line above	Use the back-arrow key '<-' to move the cursor to the right-hand end of the previous line.
Create work space	Use SHIFT/CLR to obtain a clear screen for writing program lines. This command does not delete existing lines, which may be scrolled into view. Line numbers are automatically placed correctly when program is listed.
Insert blank line	Press 'f1' followed by INST to insert a blank line at the cursor position.
Copy line	Change the line number to the new line number. The old line will still exist.
Delete line	To delete a program line, type the line number and press RETURN.
List program	Type the command 'list' or 'prog', without a line number, and press RETURN. You can also type 'prog <line number>' to see a listing from the entered line number.
Run program	Type the command 'run' or 'execute', without a line number, to run the program. You can also type 'run <line number> to start execution at the specified line number. At the end of the program you will

be returned to the Main Menu unless you specify the instruction 'PROG' as the last line of the program, which will return you to the Program Writer.

Invalid line

All lines entered must have a line number except when using the direct commands 'list', 'prog', 'run', and 'execute'. The error message 'Invalid line: Re-enter' usually indicates a missing line number or a command syntax error. The highest line number allowed is 63999.

Check available memory

Up to 4K is available for programs and variables. You can find out how much is left by returning to the Main Menu and typing 'display fre(0)'. The number displayed is the number of bytes left. For complete accuracy, type 'clr' first to clear out the program variables.

Print program listing

The best way to obtain a hard copy program listing is to return to the Main Menu and type 'print:list:display'.

1.3 CREATING A PROGRAM

In the sections which follow the creation of programs will be explained and you will be introduced to some elementary programming techniques.

It will be explained how a program is developed out of a sequence of COMMAND LINES and how such devices as LOOPS and CONDITIONALS can be utilized to make your programs more versatile.

If you are unfamiliar with the commands used in these sections which do not appear on the Main Menus you should refer to the section on ADDITIONAL COMMANDS, Programming Section 2.

1.3.1 PROGRAMS AND COMMAND LINES

In Section 3 the notion of a COMMAND LINE is explained. A Command Line is simply one or more Superbase commands separated by colons (:).

A PROGRAM is essentially nothing more than a sequence of such Command Lines where each has been given a LINE NUMBER determining the order in which the instructions are to be carried out.

In fact Programs can be created without the use of the PROG Option at all, simply by typing in Command Lines from either Main Menu as normal, but with a line number in front of them as in:

```
100 find "update-list" where [update] is "=yes"
```

This could then be followed by another such as:

```
200 output the records from "update-list"
```

All we need to make these two lines into a simple program is another line to signify that the program has ended:

```
300 end
```

The reason for numbering lines in hundreds is to allow plenty of room for further lines should you want to edit the program later with the PROG Option.

Having typed in a sequence of numbered Command Lines in this way you then have a program resident in memory which can be executed with the 'f7' key from Menu 1, or directly with the (unnumbered) command 'EXECUTE' from either Main Menu, or with the command 'RUN'.

Programs are only held in the computer's memory until the power is switched off unless they are stored on disk with the SAVE command.

To store a program you have created in the PROG Option, use the SAVE command from the Main Menu, as described above.

1.3.2 PROGRAM LOOPS

The program created in the last section simply constructs a Key List of records that need updating and displays them. Suppose however that we wanted to actually carry out operations on these records such as adding the contents of one field to the contents of another.

We might have a field called 'Last-paid' which we want to add to the contents of a field called 'Total-paid', as in the CUSTOMER INVOICE file of tutorials 1 and 2.

The Command line for this calculation is:

```
calc [Total-paid]=[Total-paid]+[Last-paid]:store the record
```

Replacing line 200 in the three line program above with this Command Line however, would only result in the calculation being performed on the current record. To carry out the operation on all of the records in the Key List requires what is known as a PROGRAM LOOP.

A Program Loop consists of two lines, the second of which returns to the first, in between these two lines will be a number of other Program Lines which carry out the operations required. These lines constitute the MAIN BODY of the loop.

For example, in the following program:

```
100 find "hlist" where [update] is "=yes"
200 select from "hlist"
300 calc [Total-paid]=[Total-paid]+[Last-paid]:store the record
400 goto 200
```

lines 200 and 400 constitute the loop itself and line 300 is the body of the loop. (Note that this program as it stands will loop indefinitely.)

Line 200 tells the computer to select a record from the Key List, line 300 specifies an operation to be carried out on that record and stores the modified version and line 400 tells the computer to go back to line 200 to select another record for processing.

In this way all of the records in the list would be processed.
A word of warning. The 'goto' command is very useful but can if overused become a source of great difficulty when debugging long programs.

1.3.3 CONDITIONALS

The only thing wrong with the above program is that once the computer enters the loop it will never get out unless the STOP key has been enabled and is used, or the power is switched off.

What is needed is a Program Line in the body of the loop which will return to the Main Menu once all of the records in the Key list have been processed.

For this purpose we can include another line-

```
350 eol Menu
```

The command 'Eol' which stands for 'End of list' is what is known as a CONDITIONAL since the command following it (in this case the MENU command which returns us to Menu 1) is only executed on the condition that EOL is satisfied. EOL is satisfied whenever the end of the list has been reached.

Any Command Line can follow EOL, as in

```
eol display "All records processed":wait
```

or

```
eol output all records from "hlist"
```

As in our original example, these Command lines will not be executed unless the end of the list has been reached.

1.3.4 OTHER CONDITIONALS

Two other useful conditionals are 'EOF' and 'IF...THEN...'

The first of these, EOF is similar to EOL except that it is satisfied (i.e. the rest of the Command Line following it is activated) when the end of the file has been reached.

EOF would be used when ALL of the records in the file were to be processed as in:

```
100   select the first record
200   [Total-paid]=[Total-paid]+[Last-paid]:store the record
300   select the next record
400   eof menu
500   goto 200
```

Another useful conditional is the BASIC command 'IF...THEN...', as in:

```
300 if [Outstanding] > [Credit-limit] then: display
    [Name];[Address];[Outstanding]
```

This Program Line would display the requested information only IF the particular customer owes more than their credit limit allows.

Any condition can follow the 'IF' and any command may follow the 'THEN', so the general format of the IF...THEN... command is:

```
if <condition> then <command>
```

However, if you use a Superbase command as opposed to a standard BASIC command after the 'THEN', you must put a colon immediately after the 'THEN':

```
if <condition> then: <Superbase command>
```

1.4 LINKING BETWEEN FILES

Another facility useful in User-written programs is the set of commands SETLINK, LINK, RLINK and ELINK which enable you to update a record with information from a record in another file.

Suppose that we wanted to add the contents of the 'Last-paid' field in the Customer Invoice file to the contents of the 'Balance' field in the Customer Records file from Tutorials Two and Three.

The following modifications to the program outlined in section 14.3.2 would suffice:

```
200   setlink "Cust.Rec"
300   find "hlist" where [update] is "=yes"
400   select from "hlist"
500   eol menu
600   [Total-paid]=[Total-paid]+[Last-paid]
700   store the record
800   a=[Last-paid]
900   link [ref]
1000  [Balance]=[Balance]+a:store
1100  rlink
1200  goto 400
```

Note that line 200 sets the link to the Customer Records file, the link remaining in force throughout the program. Line 800 stores the amount last paid in a BASIC variable. Line 900 links to the record in that file having the same Customer Reference as the record we are linking from. Line 1000 adds the value stored in the BASIC variable to the Balance field in the record we have linked to. Line 1100 returns from the file that was linked to, and line 1200 returns us to select the next record key from "hlist".

We could improve this updating program with the following lines:

```
   100 file "Cust.Inv"
```

This ensures that when the program is executed, the correct file is used even if the current file is something else.

```
    650  [update]="no"
```

This line modifies the Update field to show that the record has been updated.

Finally, note that in Programs you can 'select' a record directly using any of the forms: 'select a$', 'select "jones"', 'select [name]', or 'select left$(a$,4)' (or any BASIC string expression).

(Editing hint: while testing a program, insert a temporary line containing the commands 'wait:prog'. This returns you to the editor automatically.)

1.5 LOADING PROGRAMS FROM DISK

By constructing programs in this way, using the PROG Option to write and modify them you can save a library of programs on disk.

Whichever program (if any) you last created since switching on the system will be the current program in memory, and this will be the program executed if you press the 'f7' key from Menu 1 or type in the command 'EXECUTE' from either Main Menu.

If you want to execute one of the other programs on disk you have to load it first so that it becomes the current program in memory.

To do this simply type the command

```
load "programname"
```

from one of the Main Menus and the program you have selected will be ready to execute.

1.6 VARIABLE AND FIELD NAME RULES

Field Names may be up to 12 characters long, with no spaces. The characters (!), (#) and (") are not allowed.

Any word may be used in a field name.

BASIC variable names may be any length, but only the first 2 characters are significant, i.e. 'elephant$' is the same as 'elegant$'.

The permitted character range is A to Z together with 0 to 9. The first character may not be a number.

No part of a variable name may contain a BASIC or a Superbase command. Examples of illegal names are:

```
do$                 newvalue$
do%                 link$
screen$             date$
total
```

A common and perplexing syntax error can be caused by inadvertent combinations of letters in program lines, for example:

```
if t and (b=3) then ...
```

produces an error because the function TAN will be detected. The solution is to use parentheses as follows:

```
if (t) and (b=3) then ...
```

1.7 PERMITTED BASIC COMMANDS

Because Superbase incorporates many high-level commands, not all the standard BASIC commands are available. A list of valid commands follows.

PRIMARY

clr	goto	rem
data	if...then	restore
dim	list	return
for...next	load	run
end	new	save
get	on	step
gosub	read	stop

OPERATORS AND EXPRESSIONS

abs	int	right$
asc	left$	sin
atn	len	sqr
chr$	log	str$
cos	mid$	tan
exp	'pi'	val
fre		

2 ADDITIONAL COMMANDS

A number of additional commands are provided which do not appear on either of the Main Menus.

These commands can either be used in COMMAND LINES typed in from one of the Main Menus (see Reference Section 3) or can feature in PROGRAMS written by the user (see Programming Section 1).

While many users will opt to use only the Menu driven commands of Superbase, the advanced user will find that the addition of the commands detailed in this section extend the power of the system considerably.

In fact, used as a command driven system, Superbase is an advanced Database Language which extends BASIC by at least 40 commands.

In the following sections a specification is given of the additional commands which includes a category titled STATUS.

The status of a command is either PRIMARY or SECONDARY. SECONDARY commands are those commands which must follow other commands and cannot be executed on their own.

2.1 THE COMMANDS

2.1.1 ACROSS

STATUS: Primary or Secondary Command

FORMAT: < <command> across "<string>";<[field]...> >

PURPOSE: To set the output to ACROSS the page or screen.

COMMENTS: See Reference Section 8.6.

2.1.2 ALL

STATUS: Secondary Command

FORMAT: output all < <format command> <[field]...> >

PURPOSE: Used with the OUTPUT command to signify that ALL records in the file are to be used for the output operation.

EXAMPLES: output all the records @20 [Name][Outstanding]

Outputs the contents of the NAME and OUTSTANDING fields from every record in the file.

2.1.3 ASK

STATUS: Primary Command

FORMAT:
```
ask x$
ask x
ask "text";x$
ask "text";x
ask [field]
ask "text";[field]
ask @x,y <all options>
ask &n <all options>   (where n is a number, 1-254)
```

DEFAULT: Enter? (in command area at the top of the screen).

PURPOSE: To allow input from the keyboard during program execution.

COMMENTS: The text following the 'ASK' command will be displayed on the screen and whatever you type will become the value of the TEXT VARIABLE (X$) or FIELD ([field]) specified. To avoid formatting problems, do not enter double quotation marks. RETURN on its own is not accepted. Non-numeric input to numeric variables or fields will produce the 'Invalid Numeric Result' error message. The same message appears if the input number is too large for a field format, or if a result field calculation overflows its format as a result of a field parameter input with 'ASK'.

The "text" need not include the word 'Enter' as this is supplied for all prompts that do not have positioning commands @x,y.

The & option must precede all others if used; the default is 40.

This command can be used to set up user-defined Menus.

EXAMPLES:
```
100  display @5,5"1> Update Accounts";@5,8"2> Show
     Clients over Credit Limit";@5,11"3> Print
     Report on Sales"
200  ask a
300  if a<1 OR a>3 then goto 200
400  on a goto 500,600,700
500  load "Program 1"
600  load "Program 2"
700  load "Sales Report"
```

Displays a Menu from which user-written programs can be accessed.

2.1.4 BRKOFF STOP KEY OFF

STATUS: Primary Command

FORMAT: brkoff

PURPOSE: To disable the STOP key after it has been enabled with the BRKON command.

2.1.5 BRKON - STOP KEY ON

STATUS: Primary Command

FORMAT: brkon

PURPOSE: To enable the STOP key to function as a means of breaking out of a program that is running.

COMMENTS: The STOP key can be disabled again with the BRKOFF command.

2.1.6 CLEAR

STATUS: Primary Command

FORMAT: clear

PURPOSE: Clears all fields allowing the current record to be used as a new record. Information can be entered into the fields with the CALC command. (see Section 11.4).

COMMENTS: After entering information into a record in this way, the STORE command must be used.

EXAMPLES: clear:calc [Name]="Davis, A";[goods]="1 CBM 64 Computer":store

Enters a new record with the details as shown.

2.1.7 CONT - CONTINUOUS PRINT

STATUS: Primary Command

FORMAT: **cont <1 or 0>**

DEFAULT: 0

PURPOSE: To specify whether or not you want Superbase to pause after each page has been printed.

COMMENTS: 'CONT' is normally switched on (1). To force the end of page break, set 'CONT' to 0. Printing may also be paused with the 'WAIT' command.

EXAMPLES: **cont 1**

Suppresses the end of page break.

cont 0

Switches end of page break back on.

2.1.8 CONVERT

STATUS: Primary Command

FORMAT: **convert n, x$**

convert [datefield], x$

PURPOSE: To convert a number or numeric expression, or the contents of a date or calendar field to a named BASIC string variable.

COMMENTS: 'CONVERT' allows dates to be changed from numeric form to an easily manipulable string form. Dates are stored internally as numbers.

EXAMPLES:
```
100  convert [Date], d$
200  dd$=left$(d$,2)
300  mm$=mid$(d$,3,2)
400  yy$=right$(d$,2)
500  print dd$;"/"; mm$; "/"; yy$
```

This example assumes European style dates and presents the date in a different format.

2.1.9 DATABASE

STATUS: Primary Command

FORMAT: database "name" <,unit number, drive number>

PURPOSE: To select a database to work in or to create a new database.

COMMENTS: If no database "name" is given, the prompt 'Enter Database Name' will be displayed. Enter a database name without quotation marks. If database "name" is given, it must be enclosed within quotation marks. Used within a Program, the command may transfer processing to another database without user intervention.

Defaults for unit number and drive number are 8 and 0 respectively.

Enter $ to the prompt 'Enter Database Name' to view the disk directory and ascertain database names. Database names must not begin with '$', and must not contain numbers.

2.1.10 DATE

STATUS: Primary Command

FORMAT: date x$, n

PURPOSE: To set the date-style to European (day month year) or American (month day year). Also returns the number of the month from x$ in n. If n=0 the date is invalid.

The date-style is set by each use of the command and date format may therefore be reversed.

COMMENTS: Use 'Date' in conjunction with 'ask' and test for date validity. Dates must be input using alphabetic month abbreviations.

EXAMPLES:
```
100 ask &7a$
200 date a$,n
300 if n=0 then:goto 100
```

2.1.11 DETAIL

STATUS: Primary Command

FORMAT: detail all/from <"text";<[field]>...>

PURPOSE: To specify the main body of detail in a User-written report.

COMMENTS: Instead of writing your own Report Programs, you could choose to have them written for you by the REPORT GENERATOR (see Reference Section 13).

EXAMPLES: detail from "hlist" [ref] [quantity] [price]

2.1.12 DISPLAY

STATUS: Primary or Secondary Command

FORMAT: display <command line>

PURPOSE: To set the direction of output to the screen as opposed to the printer, and optionally display any parameters following it. The command 'display chr$(147) chr$(7)' will clear the screen and sound the tone.

COMMENTS: See Reference Sections 8.2 and 8.3.

2.1.13 DO

STATUS: Primary Command

FORMAT: do "<string>"
do X$

PURPOSE: To execute a Command Line stored in a string variable or enclosed in quotation marks.

COMMENTS: A number of Command Lines can be assigned to string variables and a program can be designed to choose between them depending on predefined criteria. Multiple clauses may not be used. The maximum length of the command string is 80 characters. 'DO' must be the last command if on a multi-clause line.

EXAMPLES: a$= "display [goods]":do a$

2.1.14 DOWN

STATUS: Primary or Secondary Command

FORMAT: < <command> down "<string>";<[field]...> >

PURPOSE: To set the output to DOWN the page, one record at a time.

COMMENTS: See Reference Section 8.6.

2.1.15 DUMP

STATUS: Primary Command

FORMAT: dump "<filename>"

PURPOSE: To print, display, or store on disk a file of the variables and their current values from the program currently in memory.

COMMENTS: A file of variables can be read back into the program at a later date by means of the SET command. If no filename is given the dump will be to the screen or printer, whichever is the current output device.

2.1.16 ELINK - END OR REVERSE LINK

STATUS: Primary Command

FORMAT: elink

PURPOSE: To reverse the link established by the SETLINK command so that the linked file becomes the current file or the opposite. No actual link is active after an 'ELINK'. The 'File Selected' indicator will show which file is the current file after the 'ELINK' by displaying '='.

COMMENTS: Used with the SETLINK, LINK, and RLINK commands.

2.1.17 ENDREPORT

STATUS: Primary Command

FORMAT: **endreport < "text";<total variables> >**

PURPOSE: To specify the text and totals you want to display at the end of a report generated by a User-Written Report Program.

COMMENTS: Instead of writing your own Report Programs, you could choose to have them written for you by the REPORT GENERATOR (see Reference Section 13).

EXAMPLES: **endreport** "Total Paid ";t0;"Total Outstanding ";t1

Will display the totals accumulated in the Total Variables t0 and t1 together with text to explain what these quantities are.

2.1.18 EOF - END OF FILE

STATUS: Primary Command

FORMAT: **eof <Command(s)>**

PURPOSE: Used in user-written programs containing the 'SELECT' command to carry out the stated action when the last record in the file has been reached.

COMMENTS: EOF is a CONDITIONAL. This means that the execution of those commands which follow it in the same line, is conditional on it being true. EOF is true whenever the end of the file has been reached.

EXAMPLES: **eof Menu**

Will return to Menu 1 when the last record in the file has been processed.

2.1.19 EOL - END OF LIST

STATUS: Primary Command

FORMAT: eol <command(s)>

PURPOSE: Used in user-written programs containing the 'FROM "list"' command to carry out the stated action when the last record in the Key List has been reached.

COMMENTS: EOL is a CONDITIONAL. This means that the execution of those commands which follow it in the same line is conditional on it being true. EOL is true whenever the end of the Key List has been reached.

EXAMPLES:

```
100   select the records from "hlist"
200   calc a$=[goods];a$
300   eol display @1,0 "Press any key":wait
400   eol menu
500   goto 100
```

After all of the records in the Key List have been processed, the program will pause until a key is pressed and will then return to Menu 1.

2.1.20 FROM

STATUS: Secondary Command

FORMAT: <command> from "<list>"

DEFAULT: "hlist"

PURPOSE: To specify which Key List is to be used in a Superbase operation.

EXAMPLES: select from "debt list"

Selects just those records whose keys appear on the Key List named "debt list".

2.1.21 IS

STATUS: Secondary Command

FORMAT: <<command> where [field] is <"condition">>

PURPOSE: Used with commands containing WHERE which operate on those records WHERE some field IS of some value or range of values.

EXAMPLES: find "Debt-list" where [Outstanding] is ">0"

Creates a Key List of records of people who owe you money.

select match where [Town] is "=Paris"

Allows you to view just those records of people who live in Paris.

2.1.22 LFEED - LINE FEED

STATUS: Primary Command

FORMAT: lfeed <1 or 0>

DEFAULT: 0 (off)

PURPOSE: To send a linefeed to the printer after each carriage return.

COMMENTS: 'LFEED' is either ON or OFF <1 or 0>.

EXAMPLES: lfeed 1

Switches the linefeed on.

lfeed 0

Switches the linefeed off again.

2.1.23 LINK

STATUS: Primary Command

FORMAT:
```
link <[field]>
link "text"
```

DEFAULT: Key field of current file.

PURPOSE: To access a record in another file of which the key has the same value as the parameter in the current record specified in the LINK command.

COMMENTS: Set which file you want to link to with the SETLINK command. If no parameter is specified, the key of the current record is used. The 'File Selected' indicator will show a '+' instead of its usual '=' when a link is active.

EXAMPLES: See the 'SETLINK' example.

2.1.24 LMARG - LEFT MARGIN

STATUS: Primary command.

FORMAT: `lmarg <number>`

DEFAULT: 1

PURPOSE: Sets the left margin for both screen display and printed output.

COMMENTS: 'LMARG' can be used before an OUTPUT command to set the margin of the output. Maximum value is 250. Use with no value to reset the default value.

EXAMPLES:
```
lmarg 20
```
Sets the left hand margin to column twenty.

2.1.25 LOAD

STATUS: Primary Command

FORMAT: `load "filename"`

PURPOSE: To load a Superbase program file from disk.

COMMENTS: The ".p" suffix may be omitted. A drive specifier may be used. If a program is loaded by another (i.e. chained), it will run automatically. Variables will be cleared.

2.1.26 MENU

STATUS: Primary Command

FORMAT: **menu**

PURPOSE: To return to Menu 1. Often used with a conditional.

EXAMPLES: **eof menu**

Returns to Menu 1 when the last record in the file has been processed.

eol menu

Returns to Menu 1 when the last record in the Key List has been processed.

2.1.27 NMAT - NO MATCH CONDITIONAL

STATUS: Primary Command

FORMAT: **nmat <command(s)>**

PURPOSE: 'NMAT' is a conditional used with SELECT KEY. This means that the execution of those commands which follow it in the same line is conditional on it being true. 'NMAT' is true whenever a select key MATCH operation is unsuccessful (key not found).

COMMENTS: See Reference Section 6.1.7 for details of the MATCH command.

EXAMPLES: **nmat display "Sorry...I couldn't find that key field":wait**

Displays the above message if no record is found matching or partially matching the criteria you have specified.

2.1.28 PDEF - PRINTER DEFINITION

STATUS: Primary Command

FORMAT: pdef <number>

DEFAULT: 0

PURPOSE: To set the printer type to be used.

COMMENTS: The start-up program "start.p" may be edited to set the printer definition.

EXAMPLES:

pdef 0	CBM Dot Matrix: sends CBM ASCII code with a cursor down character at the start of each line to produce lower case letters. Graphics not available.
pdef 1	Epson MX80 type: sends true ASCII.
pdef 2	Daisywheel type: e.g. Diablo, Qume, etc., sends true ASCII.
pdef 5	Epson MX80 type: sends CBM ASCII.
pdef 6	Daisywheel type: e.g. CBM 6400, sends CBM ASCII.

Low cost daisywheel type printers should work with either 2 or 6. RS232 interface and/or printer buffer may be necessary. For non-CBM dot matrix printers, select 1 if you want Superbase to convert true ASCII, or 5 if you are using an interface device to convert.

2.1.29 PDEV - PRINTER DEVICE NUMBER

STATUS: Primary Command
FORMAT: pdev <parameter1> <parameter2><parameter3>
DEFAULT: 4,7,0 (for 1515/1525/MPS 801 printers)
PURPOSE: To set parameter 1 to device number, and if necessary parameters 2 and 3 to required values.
COMMENTS: pdev x,y,z sends to the secondary address specified in y, and so should be able to set up printers not emcompassed below. We suggest you incorporate your defaults into program "start.p"

EXAMPLES: This table illustrates appropriate pdev values:

	IEEE	RS232	CENTRONICS
64	4,7,0/4,255,4 (a)	2,8,0 (b)	0
700	4,255,0	2,8,0 (b)	0
8096	4,255,0	4,255,0 (d)	0

(a) 4,7,0 for printers requiring no interface;
4,255,0 for printers requiring interface,
e.g. 3022,4022,8023,6400.
(b) Check Appendix C for values.
(c) All connected to user port.
(d) External RS232 interface attached to IEEE port.

2.1.30 PLEN - PAPER LENGTH

STATUS: Primary Command

FORMAT: plen <number>

DEFAULT: 66

PURPOSE: To set the length of paper for reports and other printed output. Maximum is 255 and minimum 3.

EXAMPLES: plen 72

Sets the standard page length for the Commodore 1515 printer.

2.1.31 PLUS

STATUS: Secondary Command

FORMAT: <command parameters> plus <parameters>

PURPOSE: For extending the parameter list of a command beyond 1 program line.

COMMENTS: The REPORT GENERATOR (see Reference Section 13) uses 'PLUS' in the REPORT PROGRAM it creates to join the responses to its prompts after the user has answered 'Y' for 'Yes' when asked 'Any More?'.

Note that PLUS can only be used in programs and not as a Direct Command. Parameters may include text within double quotation marks. 'PLUS' is valid in DISPLAY, PRINT, OUTPUT, TITLE, TOTAL, DETAIL, ENDREPORT, and SUBTOTAL.

EXAMPLES:
```
100  display "this is a string of text too long to
     fit in the seventy nine "plus
200  "characters allowed for command lines"
```

Displays the two pieces of text as though they had been entered as one line.

2.1.32 PMAT - PARTIAL MATCH CONDITIONAL

STATUS: Primary Command

FORMAT: pmat <command(s)>

PURPOSE: 'PMAT' is a conditional used with SELECT KEY. This means that the execution of those commands which follow it in the same line is conditional on it being true. 'PMAT' is true whenever a select key MATCH operation is PARTIALLY successful, and untrue if a full match is obtained.

COMMENTS: See Reference Section 6.1.7 for details of the MATCH command.

EXAMPLES: pmat display [Name]

Displays any name which partially matches the criteria specified.

2.1.33 PRINT

STATUS: Primary or Secondary Command

FORMAT: print <command line>

PURPOSE: To set the direction of output to the printer as opposed to the screen, and optionally print any parameters following it.

COMMENTS: The 'print' command is used primarily when switching output or printing variables, field names etc. It must be preceded by 'output' when using the 'output' option syntax. See Reference Section 8.4.

2.1.34 PROTECT

STATUS: Primary Command

FORMAT: protect "<program name>"

PURPOSE: To save the current program in a form that prevents it from being listed or edited. Always take a reserve copy first.

COMMENTS: Programs protected by this command can be LOADED, RUN, or DELETED, but not LISTED or edited. Protected programs are encrypted. If an attempt is made to LIST a protected program, the message 'End of Program' will be displayed unless the first statement in the program is a 'REM' statement, in which case text following the 'REM' on the same line will be displayed. This allows Copyright warnings, etc., to be included in users' applications, which may be marketed as Superbase applications modules. Contact Precision Software for further details of marketing agreements.

2.1.35 QUIT

STATUS: Primary Command

FORMAT: quit

PURPOSE: Used to exit from Superbase and return to CBM BASIC.

COMMENTS: 'Quit' shuts down Superbase in an orderly manner, resetting the computer for other use.

2.1.36 RESTART

STATUS: Primary Command

FORMAT: restart

PURPOSE: To re-initialize the current database and file after severe system errors such as disk errors, I/O error, 'file not open' error, etc.

COMMENTS: Restart can be used to recover a deleted file if used immediately after a 'File Deleted' message caused by the SELECT DELETE command.

2.1.37 RLINK - RETURN FROM LINK

STATUS: Primary Command

FORMAT: **rlink**

PURPOSE: To return from the linked file to the original file.

COMMENTS: Used with the SETLINK, LINK and ELINK commands.

EXAMPLES: See the 'SETLINK' example.

2.1.38 RMARG

STATUS: Primary Command

FORMAT: **rmarg** <number>

DEFAULT: 80 for the printer. Screen width is fixed at 40.

PURPOSE: To set the right margin for screen display and printed output.

COMMENTS: Can be used before the OUTPUT command to set the right margin of the output. Maximum value is 255 and minimum 20.

EXAMPLES: **rmarg 76**

Sets the right margin to column 76.

2.1.39 SAVE

STATUS: Primary Command

FORMAT: **save "filename"**

PURPOSE: To save a Superbase program file on disk.

COMMENTS: A ".p" suffix is appended to the filename automatically. A drive specifier may be used. Any existing program of the same name will be overwritten.

2.1.40 SCREEN

STATUS: Primary Command

FORMAT: screen <number>

DEFAULT: Screen 0

PURPOSE: To set the default screen. This is the screen first displayed in the ENTER, SELECT and FIND options.

COMMENTS: Any of the four screens in use can be set as the default screen. The range is 0 to 3.

EXAMPLES: screen 2

Sets the default screen to the third screen.

2.1.41 SET

STATUS: Primary Command

FORMAT: set "<filename>"

PURPOSE: To execute commands or retrieve variable values stored in a sequential file on disk. Uses include the storage of control values such as incrementing transaction numbers, and the execution of sub-routines or command sequences from within a program. The filename may be a BASIC string variable.

COMMENTS: This allows you to set variables as they were when stored. The file may be stored with 'DUMP' or created with 'MEMO'. If the sequential file is created with 'MEMO', no line numbers are required. Any commands in the file will be executed in sequence without the file being loaded from disk. This means that certain programming operations may be executed with virtually no memory overhead for the program instructions. If this technique is used, do not attempt to produce a Key List, read another file or list, load or save a program, or execute a goto, gosub, or for/next loop. However, you can read the database. 'SET' must be the last command if on a multi-clause line.

2.1.42 SETLINK

STATUS: Primary Command

FORMAT: setlink <"filename">

DEFAULT: Current file

PURPOSE: To establish which of the other files in the database is to be accessed by the LINK command.

COMMENTS: The SETLINK, LINK, RLINK and ELINK commands enable information to be exchanged between files in the same database. They also allow you to access a record in another file by linking via its key.

They are particularly useful for updating one file with information from another, such as updating a BALANCE field in an accounts file with information from an invoices file.

EXAMPLES:

```
100   file "Cust.Inv"
200   setlink "Cust.Rec"
300   find "update list" where [update] is "=yes"
400   select the records from "update list"
500   eol menu
600   [update]="No"
700   [Total-paid]=[Total-paid]+[Last-paid]
800   X=[Last-paid]
900   store the record
1000  link [No.]
1010  [Balance]=[Balance]+X
1020  store the record
1030  rlink
1040  goto 400
```

For an explanation of this program, see Programming Section 1.

2.1.43 SPACE

STATUS: Primary Command

FORMAT: space <number>

DEFAULT: 0

PURPOSE: To set the number of spaces between lines in output produced by the REPORT or OUTPUT commands.

COMMENTS: 'Space 1' gives double spacing, 'Space 2' gives triple spacing etc. Range is 0 to 3.

EXAMPLES: space 1

Double spaces your output or reports.

2.1.44 STORE

STATUS: Primary Command

FORMAT: store

PURPOSE: To make permanent any changes made to a record by means of any command other than ENTER or BATCH.

COMMENTS: Unless STORE is used, changes to records made other than by ENTER or BATCH will only remain in effect until another record becomes the CURRENT record.

EXAMPLES:

```
100  find "hlist" where [Discount] is "=10%"
200  select from "hlist"
250  eol menu
300  display @1,2 [Goods];[Discount]
400  ask @5,5 "Change the discount?";a$
500  if a$<>"y" then 200
600  ask @5,7 "Enter new discount";[discount]
700  store the record: goto 200
```

Will store any changes you make to the discount rate on any goods.

2.1.45 SUBTOTAL

STATUS: Primary Command

FORMAT: **subtotal [field] "subtotal text"**

PURPOSE: To specify a field for which a change in the contents will produce a subtotal break in a report generated by a user-written REPORT PROGRAM.

COMMENTS: The 'SUBTOTAL' command is only to specify the field for the subtotal break, and any accompanying text. Subtotal break fields are only relevant if the report is using a Key List previously sorted on this field. Subtotal variables are specified with the 'TOTAL' command. To ensure that a subtotal is cleared, place it in the first line of the 'SUBTOTAL' command (see Reference Section 13).

EXAMPLES:
```
400 subtotal [Department]"Subtotal for department:"
    plus s0
500 @40 &6,2 s0
```

Prints the subtotal variable s0 and then clears it.

2.1.46 TITLE

STATUS: Primary Command

FORMAT: **title "<report title>"**

PURPOSE: To determine the text to appear at the top of a REPORT page; may include column headings.

COMMENTS: For use in User-written REPORT PROGRAMS.

Instead of writing your own report Programs, you could choose to have them written for you by the REPORT GENERATOR (see Section 13). Positioning is done with the @x,y command. 'PLUS' is used to extend titles beyond a single program line.

EXAMPLES: **title @30 "Sales Report 18/10/83"**

2.1.47 TLEN - TEXT LENGTH

STATUS: Primary Command

FORMAT: tlen <number>

DEFAULT: 60

PURPOSE: To set the number of lines of text to be printed on the page. Maximum is 255 and minimum 3.

EXAMPLES: tlen 55

Sets the number of lines of text to be printed on the page to 55.

2.1.48 TOTAL

STATUS: Primary Command

FORMAT: total <total parameters> <subtotal parameters>

PURPOSE: To specify totals and subtotals to appear in a report generated by a User-written REPORT PROGRAM.

COMMENTS: The parameters are calculations on Total Variables (T0, T1, ... T9), and Subtotal Variables (S0, S1, ... S9), and have the same format as calculations in the CALC Option.

Instead of writing your own report Programs, you could choose to have them written for you by the REPORT GENERATOR (see Section 13).

EXAMPLES:
```
400 total t0=t0+[total-paid] plus
500 t1=t1+[outstanding] plus
600 s0=s0+[total-paid]; s1=s1+[outstanding]
```

At the end of the report total paid and amount outstanding will be available for printing. At each change of the field or fields specified in 'SUBTOTAL' any Subtotal Variables specified are available for printing.

2.1.49 WAIT

STATUS: Primary Command

FORMAT:
```
wait
wait a
wait a$
```

PURPOSE: Used within a user-written program to force the program to pause until a key is pressed.

COMMENTS: 'Wait' is useful when a program is displaying information on the screen. It returns a single keystroke from the user - no additional RETURN is needed. 'Wait a' will accept only numeric input.

EXAMPLES:
```
100 'find "hlist" where [town] is "=London"
200 select from "hlist"
300 eol menu
400 wait
500 display [Name];[Address]
600 goto 200
```

The program will pause between each record it retrieves until a key is pressed.

2.1.50 WHERE

STATUS: Secondary Command

FORMAT: <Command> "list" where [field] is "<condition>"

PURPOSE: Used with FIND and SELECT MATCH to specify a condition to be met by those records to be selected.

COMMENTS: See Reference Sections 7 and 6.1.7 for details of the FIND and SELECT MATCH commands.

EXAMPLES: find "update list" where [Pending-Update] is "=yes"

Creates a Key List of those records which need updating.

Select Match where [Name] is "=Jones"; [City] is "=London"

Displays all records belonging to people named Jones who live in London.

TECHNICAL APPENDIX

The Technical Appendix provides information for both the general user and the applications developer. The four sections are (1) Maximum System Values; (2) Data Storage; (3) File Manager; (4) Start-up Program.

Note: "Unlimited" in the descriptions below implies limited by disk storage constraints only.

MAXIMUM SYSTEM VALUES

Database Level	Database Name:	16 characters
	Databases:	unlimited number
	Files in a database:	15
	Programs in a database:	unlimited
	Key Lists in a database:	unlimited
	Memo files in a database:	unlimited
File Level	File names:	16 characters (key list, memo) 14 characters (program) 10 characters (data)
	Records in a file:	unlimited
	Linked files at one time:	1
	Memo file length:	23 screen lines
	Program size:	4k including variables
	Key List length:	unlimited (may be appended)
Record Level	Fields:	127, including key
	Descriptive text in file definition:	1k
	Key - anywhere in record:	1 field
	Combined result, calendar and constant fields:	32 fields
	Record length:	1108 characters
	Screens:	4
	Calculations:	79 characters expanded, 30 characters compressed (Field names take 2 characters. Functions including '(' and ')' take 1. Spaces do not count).
Field Level	Field name length:	12 characters
	Key:	30 characters
	Text:	255 characters
	Numeric:	9 digits, up to 4 after decimal pt, plus 1 sign character position
	Date:	7 or 11 chars. - 1Jan1900 - 31Dec1999
	Calendar:	7 or 11 characters
	Constant:	30 characters
	Result:	as numeric
Miscellaneous	Command line:	79 characters
	Program line:	79 characters
	Screen width:	40 characters
	Screen length:	23 lines
	Printer columns:	255

DATA STORAGE

When calculating **maximum** stored record length, follow these rules:

1. Count every character position visible between field angle brackets for text, key, numeric, result, and constant fields. Allow 5 characters for each date or calendar field. Replica fields do not count.

2. Add 1 byte for field separators (i.e. number of fields minus one). Replica fields do count. Separator is chr$(96).

3. Field names and descriptive text do not count.

4. Trailing spaces and numeric zeros are not stored. Nor are leading spaces.

5. Add 1 record separator chr$(0) per record.

6. Floating point storage, which requires minimum 5 bytes per number, is not used, allowing more economical storage of integers < 10000.

7. Database files are linked into the main disk directory allowing the use of the disk VALIDATE command from outside Superbase.

When calculating file length, follow these rules:

1. Minimum data area is 128 bytes per record, 5 bytes of which is required by the system, leaving 123 bytes for data. If more than 123 bytes are used in a record then a further 128 bytes will be assigned. This is a design feature that optimizes record access times and efficient dynamic disk space allocation.

2. File indexes are stored as 1 key per record (trailing spaces removed) plus 3 bytes. Index size varies according to the length of the key (the shorter the better) and the number of blocks in use. Blocking factor (average percentage of index block in use) is approximately 65%. As the index grows it creates a multi-level tree structure with higher levels pointing to index blocks instead of data areas as in the bottom level (B+ tree type).

3. Database files and data will be stored on the same drive. File definitions should be stored on the same drive as the database but could be accessed from another drive. Key lists, help screens, programs, import/export files, and 'output to' files may be stored on either drive (use the '1:' prefix if necessary). Output to a named list, program, memo or export file will overwrite any existing file of the same name.

4. Data is stored from and including track 2 of the disk. 5k is therefore reserved for non-database storage. Directory areas of the disk will be avoided.

FILE MANAGER

Superbase requires approximately 32k of memory for program code. 4k is allocated for user programs, but chaining is possible (see also the SET and LOAD commands).

More than 1 file definition may reside in memory at any one time, provided that the total number of fields does not exceed 127, the total number of calculations does not exceed 32, and the total amount of descriptive text does not exceed 1k. A maximum of 3 simultaneously resident definitions is possible if these totals are not exceeded. Multi-file applications will run faster if file definitions do not have to be loaded from disk each time the current file changes with the FILE command.

The process of creating a file clears all definitions from memory.

START- UP

A start-up program, "start.p", is included on the Superbase disk. It is optional, and if not present at start-up may be ignored by pressing RETURN when the "File Not Found" message appears. The original start-up program resides on the Superbase disk. It may be edited using the PROG Option to suit each user's requirements. Applications developers may modify the program to present applications of their own design.

"Start" may for example display HELP screens containing program option menus, request and validate input (including passwords), and load programs for execution.

Systems do not have to use the standard Superbase Option menus at all.

Also if "start" and other programs are encrypted with the PROTECT command, a completely secure application can be created.

An annotated listing of "Start" is included here. Study it to understand how Superbase's system parameters may be set. These parameters are set automatically to the values below at load time but have been included in the program as an explanatory aid:

```
brkon          STOP key enabled
across         OUTPUT command direction mode
display        OUTPUT command device mode
lmarg 1        Left printer margin 1
rmarg 80       Right printer margin 80
plen  66       Paper length 66 lines
tlen  60       Text length 60 lines
lfeed 0        Linefeed off
cont  1        Continuous print on
space 0        Single line spacing
pdev  4        Output device no. 4
pdef  0        CBM printer type
```

THE START PROGRAM

```
100  brkon:gosub 400:rem allow break key and set system parameters
110  display @11,3"-------------"
120  display @11,4"["@+"Superbase 64""]"
130  display @11,5"-------------"
140  display @1,17"-----------------------------------------------"
150  display @1,19"-----------------------------------------------"
160  display @0:rem reset display count
170  ask &16 @1,18"   Enter Database Name :";a$
180  database a$,8,0:rem disk 8,drive 0
190  file:rem superbase asks for file
200  new:rem clear program goto menu
400  rem *** set system parameters ***
410  lmarg 1:rmarg 80:rem margins
420  plen 66:tlen 60:rem page & text length
430  pdev 4:pdef 0:rem printer device 4 cbm code
440  lfeed 0:cont 1:rem no line feeds, continuous print
450  space 0:across:rem normal line spacing, output across
460  screen 0:rem screen 0 is default
470  return
```

ERROR MESSAGES

Note: ERROR LIGHT Superbase uses the error channel extensively when allocating disk space. Error light flashing during disk write operations does not normally indicate disk errors.

ALREADY LINKED

A LINK command has been issued while a link is already in progress.

COMMAND SEQUENCE ERROR

You have entered a Command Line having a Command in the wrong position.

DATA MISMATCH

System error or hardware failure. Export all files, create a new database using same file definitions and import the data. Can be caused by failure to use 'database' command after changing disks. Enter 'database' RETURN and carry on.

DATABASE NOT FOUND

The DATABASE you have selected is not on the current disk.

DATABASE NOT SELECTED

Commands have been issued that require access to a DATABASE when none is selected.

DISK ERROR MESSAGES

Standard COMMODORE disk error messages are displayed on the STATUS LINE, or the top line of the screen. See your disk drive manual for further details.

Certain errors are also detected by Superbase because of device malfunction. These are input/output errors and are displayed as:

I/O Error No.		
	0	Routine terminated by the STOP key
	1	Too many open files
	2	File already open
	3	File not open
	4	File not found
	5	Device not present
	6	File is not an input file
	7	File is not an output file
	8	File name is missing
	9	Illegal device number

EQUATION ERROR

You are either attempting to assign a value to a field of the wrong type to accept that value (e.g. attempting to assign a string of TEXT to a NUMERIC field), or else you have missed out an 'IS' or '=' where there should be one.

FIELD NAME OR BRACKETS ERROR

Your Command Line either refers to a non-existent field or you have typed an uneven number of brackets in the Command Line.

FIELD TOO LONG

You have entered a calculation which assigns a string too large for the field you have assigned it to.

FILE DEFINITION INVALID

The FILE command has attempted to load a file definition and accessed an invalid file. This error can occur if the file definition is overwritten by a key list of the same name. To recover, reformat the file using 'FORMAT'.

FILE DELETED

You have issued a SELECT DELETE command to a file with no records, which has caused the Database to delete the file. See the RESTART command.

FILE NOT FOUND

The FILE you have selected is not on the current disk.

FILE NOT SELECTED

You have attempted to issue a command requiring use of a file before selecting a file.

FMS COMMAND ERROR

The File Manager has detected an invalid command due to system or hardware error. Reload Superbase if you cannot continue work.

FORCED FIELD: PLEASE ENTER DATA

You are attempting to store a record/leave a KEY FIELD or another field, that has been set as a FORCED FIELD empty of information.

FULL DISK

There is no more space available on the disk you are using.

INDEX MISMATCH

System error or hardware failure. Export all files, create a new database using same file definitions and import the data. Can be caused by failure to use 'database' command after changing disks. Enter 'database' RETURN and carry on.

INSIDE FIELD: CAN'T SET

You are attempting to set a field in a position already occupied by another field.

INVALID COMMAND PARAMETER

You have issued a command with a parameter too large, too small, or of the wrong type.

INVALID DATE

You have attempted to enter into a DATE FIELD information not of the form '05may83' or 'may0583'.

INVALID DIRECT COMMAND

You have issued a command which is either a command only for use within a PROGRAM or is totally foreign to Superbase.

INVALID FMS PARAMETER

You have tried to access a record by using an INVALID KEY. Can be caused by editing a KEY LIST or using an invalid or corrupt KEY LIST.

INVALID LINE: RE-ENTER

You have tried to enter a PROGRAM LINE with no LINE NUMBER while in the Program Writer provided by the PROG Option.

INVALID LINK FILE

You have attempted to LINK to a non-existent file or to a file in a different DATABASE.

INVALID NUMERIC RESULT

You have either assigned a value too large or of the wrong type to a numeric field by means of an ASK or CALC command, or the result of your using one of these commands has caused a RESULT field to assume a value of the wrong type or too large for its format.

INVALID SCREEN NUMBER

You are trying to go to a screen which has not been formatted for the current file, or to format a screen when the current file has already a full complement of four screens.

KEY ALREADY EXISTS

You are trying to ENTER a record with a KEY which already exists on another record.

KEY FIELD NOT DEFINED

You have tried to format a RECORD LAYOUT without including a KEY FIELD.

LINE NOT FOUND

You have issued a GOTO command to a Program Line that does not exist.

NO FIELD DEFINED

You have attempted to format a RECORD LAYOUT without any fields at all.

NO HELP AVAILABLE

You have requested a HELP SCREEN that does not exist.

NO LINK SET

You or a Program have issued a LINK command without having first specified which file is to be linked to with the SETLINK command.

NO PROGRAM PRESENT

You have issued an EXECUTE command when there is no Program resident in memory.

NOT A DATABASE FILE

You have attempted to select a database with a filename which is that of a file other than a database file.

NOT A PROGRAM FILE

You have specified a filename in a LOAD or EXECUTE command which is not the name of a PROGRAM but of some other file.

OUT OF MEMORY

Your computer has run out of free memory space.

RECORD TOO LONG

You have tried to format a record larger than the permitted size (for limitations on record sizes see the TECHNICAL SUMMARY).

SEMICOLON MISSING ERROR

You have omitted a semicolon between calculations in a CALC command.

SYNTAX ERROR

Incorrect use of Superbase and/or BASIC commands or functions. It may help to look at Programming Section 1.6.

TOO MANY COMMENTS

You have attempted to FORMAT a RECORD LAYOUT containing too much DESCRIPTIVE TEXT (for limitations on amounts of descriptive text see The TECHNICAL APPENDICES)

TOO MANY FIELDS

You have attempted to FORMAT a RECORD LAYOUT containing too many fields (for limitations on numbers of fields see The TECHNICAL APPENDIX.

TOO MANY FILES

You have tried to create more than the maximum fifteen files for a given database.

RS232 CONTROL REGISTERS

Users of the RS232 port will need to set the values of 2 CONTROL REGISTERS in order to instruct the computer how to send data to the printer. These values are entered with the command 'PDEV 2,x,y' where x is the value described in 3 and y is the value described in 4 below.

1. Study your printer manual and note down the correct settings for these items:

   baud rate
   data word length
   number of stop bits
   parity

2. For each of these items, look up the corresponding value for the required setting in the appropriate table below, and note the value down.

BAUD RATE	VALUE
50	1
75	2
110	3
134.5	4
150	5
300	6
600	7
1200	8
1800	9
2400	10
3600	11
4800	12
7200	13
9600	14
19200	15

DATA WORD LENGTH	VALUE
8	0
7	32
6	64
5	96

NO. OF STOP BITS	VALUE
1	0
2	128

PARITY TYPE	VALUE
disabled	0
odd	32
even	96
mark transmit	160
space transmit	224

3. Add together the values you obtained for baud rate, data word length, and number of stop bits. Enter this as 'x'.

4. Enter the value you obtained for paritv as 'y'.

Example

```
baud rate 9600        - value 14
word length 8         - value 0
no. of stop bits 1    - value 0

total value of A, B and C - 14

Enter 14 as 'x'

parity odd            - value 32

Enter 32 as 'y'
```

You can incorporate these values into the start-up program "start.p" as defaults. See Appendix A the Technical Appendix.

SELECTED GLOSSARY

BASIC VARIABLES

Locations in the computer's memory which can be used to store information temporarily during processing.

BORDER

A string of characters, usually graphics characters, used to draw a border around the SCREEN LAYOUT.

CHARACTER

A single symbol that appears on the screen. Created by pressing a key or combination of keys.

COMMAND AREA

The top two lines of the screen where commands are entered. This is also the MESSAGE AREA where Superbase displays its messages to you.

COMMAND LINE

A sequence of commands separated by colons and executed one after the other.

CURSOR

The flashing block on the screen that indicates where the next character which is typed will appear.

CURSOR CONTROL KEYS

Marked with arrows. Used to move the cursor.

DATA

Information stored in FILES.

DATA DISK

The disk used for storing your files.

DATABASE

A collection of up to 15 Files held together on disk.

DEFAULT LIST

The KEY LIST created during a FIND or SORT operation if no listname is specified. The default name "hlist" is assigned.

DEFAULT VALUES

Those values which Superbase assigns to parameters if no others are specified.

DELETE

Remove text without leaving a gap, i.e. the remaining characters move up to fill the space that had been occupied by the deleted text. (Contrast with "erase".)

DESELECT

The operation of leaving a selected option without entering any parameters or taking any other action.

DESCRIPTIVE TEXT

Text that appears on a SCREEN LAYOUT to make clear what the various items in the record are meant to be. This may include graphics characters to underline or box in sections of the screen or to provide a BORDER around the screen.

DIRECTORY

A list of the names of the files on a disk.

DISK FORMATTING

The process of setting up a blank or recycled disk for use as a Superbase DATA DISK.

ERASING

Blanking a line of DESCRIPTIVE TEXT with spaces without filling the gap left by the line erased (as opposed to DELETING).

FIELD

A 'blank slot' where information is stored in the record.

FIELD END MARKER

A striped rectangle which appears in the Format Option to signify where a field ends.

FILE

A collection of Records stored together on disk with the same Screen Layout.

FILE DEFINITION

The file on disk storing the details of the RECORD LAYOUT.

FILL FILE

A file of information used by a wordprocessing program to fill in blank slots (variable blocks) in a standard letter.

FORCED FIELD

A FIELD which must have data entered into it and may not be left blank.

FORMATTING COMMANDS (1)

Commands which enable you to adjust the way that output is displayed on screen or printed on paper.

FORMATTING COMMANDS (2)

Commands which are used to set up the structure of the screens in your RECORD LAYOUT.

FUNCTION KEYS

The large keys on the right of the CBM 64 keyboard, used to control many Superbase operations.

HOME POSITION

The top left corner of the screen or text. The cursor can be moved to the top left of the screen by pressing CLR/HOME.

INSERT

Add characters, words or lines in between other characters, words or lines in a descriptive or other text.

INVERTING

Changing from dark characters on a light background to light characters on a dark background. Single lines or whole screens can be inverted.

ITEM LIST

A list of parameters following an OUTPUT, SORT or BATCH command. It may include field-names and/or BASIC variables and expressions.

KEY

Every record has a Key, which is whatever is stored in the Key Field of the record. The Keys are what Superbase uses to order the records in the file.

KEY LIST

A list of KEYS of records which can be used to restrict various Superbase operations to just those records whose keys appear on the list.

LINE NUMBERS

Numbers placed at the beginning of COMMAND LINES to turn them into PROGRAM LINES.

LOAD

Take a file that is on a disk and copy it into the computer's memory.

MATCH CRITERIA

The information you enter into a RECORD TEMPLATE to determine which RECORDS are to be selected for viewing or for incorporating into a KEY LIST.

MEMORY

That part of the computer where your document is held. The contents of memory are erased when the computer is turned off.

MESSAGE AREA

The top line of the screen where Superbase displays its messages to you. This is also the first of the two lines where you can enter COMMAND LINES.

PARAMETER

A named field, named BASIC variable, string or numeric expression that may include field names or BASIC variables, or number, entered as part of a command to tell the program what information to process, and sometimes how to process it.

PROGRAM

A sequence of numbered COMMAND LINES which are executed in numerical order.

PROGRAM LINE

A COMMAND LINE which has a number in front of it so that it can be grouped with other program lines to form a PROGRAM. These Command Lines are then executed in the order dictated by the LINE NUMBERS.

RECORD

A collection of FIELDS which can be spread over up to four screens and can be regarded as a unit. Records are held together in FILES and every record within a file is of the same size and layout and has the same fields as the others in that file.

RECORD FORMAT

The SCREEN LAYOUTS of the RECORDS in a FILE.

RECORD LAYOUT

See RECORD FORMAT.

RECORD TEMPLATE

The blank RECORD FORMAT as provided to enable you to enter MATCH CRITERIA or criteria to determine which RECORDS are to be included in a KEY LIST.

REPLICA FIELD

A record field copied from an already defined field in the same record.

SAVE

Take a file that is in the computer's memory and copy it onto a disk.

SCRATCH

Remove a file from a disk.

SCREEN LAYOUT

The skeleton of a record screen consisting of FIELDS and DESCRIPTIVE TEXT. This is a blank record form like an empty card in a card-index.

SCREEN DUMP

A printout of whatever is on the screen at any given time.

SEQUENTIAL FILE

A file which consists of a stream of data with each field separated by a RETURN. Sequential Files can be used to transfer data between Superbase and other programs or between separate Superbase DATABASES.

SORT PARAMETERS

The parameters entered in the SORT Option which determine the order in which the records are to be sorted.

STRING

A sequence of characters (letters and/or numbers).

STORE

Save a file from the computer's memory onto a disk.

TRUNCATION

Trimming or shortening a string of characters to a particular size.