

Introduction to DATATRIEVE-11

Order No. AA-X025A-TK

September 1983

This document introduces you to basic DATATRIEVE-11 commands

OPERATING SYSTEM AND VERSION: RSX-11M
RSX-11M-PLUS
RSTS/E

SOFTWARE VERSION: DATATRIEVE-11 V3

digital equipment corporation, maynard, massachusetts

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by DIGITAL or its affiliated companies.

Copyright © 1983 by Digital Equipment Corporation. All Rights Reserved.

The postage-paid READER'S COMMENTS form on the last page of this document requests your critical evaluation to assist us in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

digital™

MASSBUS

PDP

P/OS

Professional

DEC

DECwriter

DIBOL

UNIBUS

VAX

VMS

VT

Rainbow

RSTS

RSX

DECmate

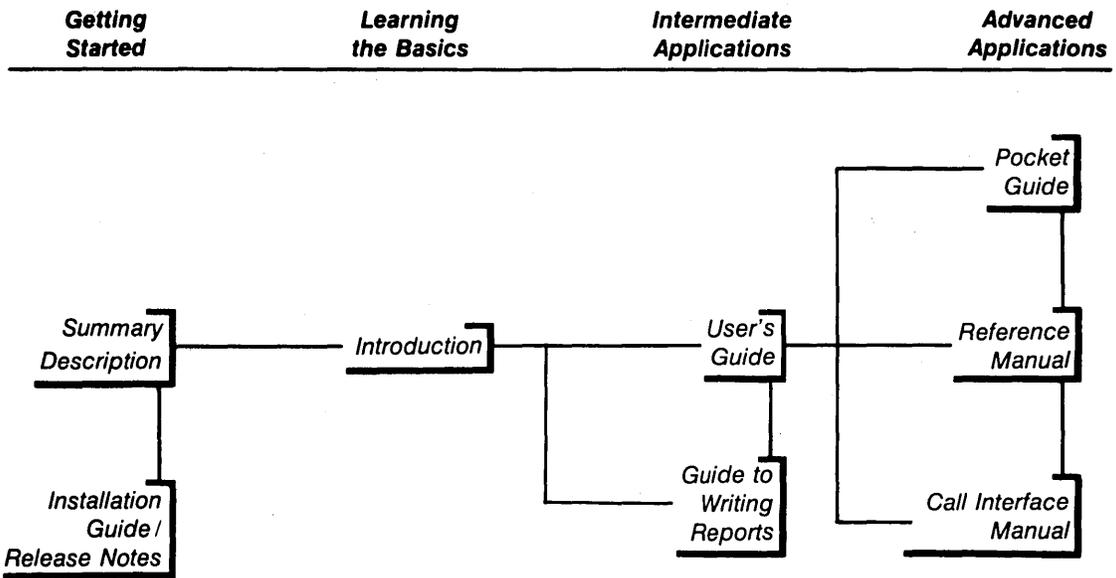
DECsystem-10

DECSYSTEM-20

DECUS

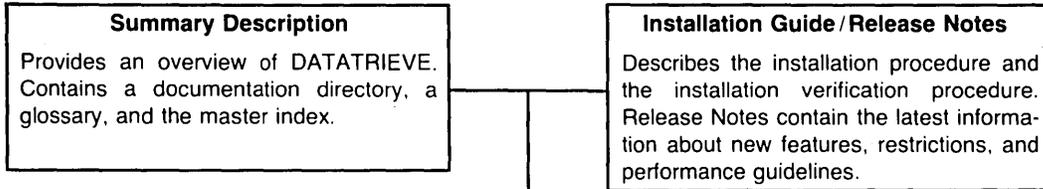
Work Processor

DATATRIEVE-11 DOCUMENTATION MAP

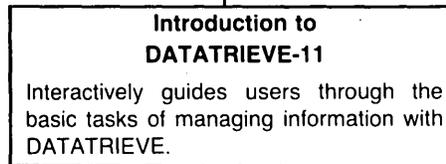


DATATRIEVE-11 DOCUMENTATION DESCRIPTION

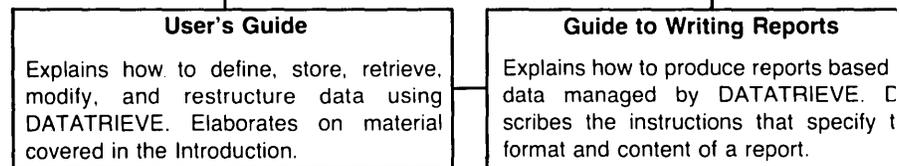
GETTING STARTED



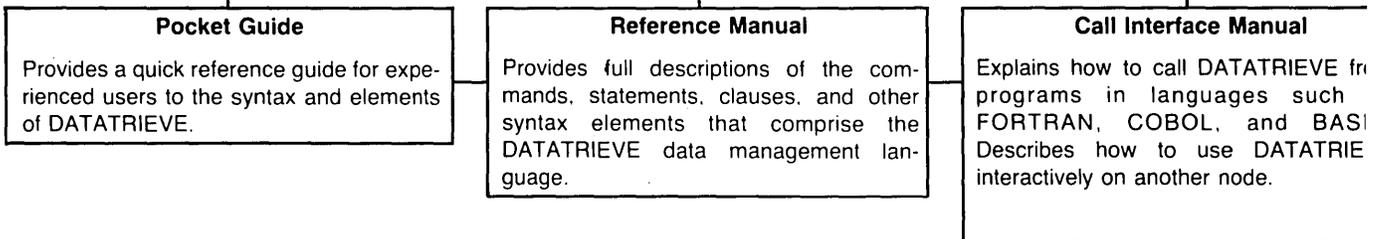
LEARNING THE BASICS



INTERMEDIATE APPLICATIONS



ADVANCED APPLICATIONS



Contents

	Page
How to Use This Manual	ix
1 Introduction	
1.1 DATATRIEVE Terminology	1-2
1.2 Using DATATRIEVE-11.	1-2
1.3 The Sample Domains	1-2
2 Starting and Stopping a DATATRIEVE Session	
2.1 Starting DATATRIEVE	2-1
2.2 Displaying Information About Your DATATRIEVE Environment	2-4
2.3 Accessing Domains	2-4
2.4 Ending a DATATRIEVE Session	2-5
3 Defining Data with ADT	
3.1 Planning a DATATRIEVE Domain	3-2
3.2 Using ADT	3-4
3.2.1 Invoking the .CMD File	3-8
3.3 Defining Domains Outside ADT	3-9
4 HELP and GUIDE Mode: Quick-Reference Section	
4.1 Using DATATRIEVE HELP.	4-1
4.2 Using GUIDE Mode	4-3
5 Storing Data	
5.1 Preparing a Domain for Storing Data	5-1
5.2 Using the STORE Statement	5-2
5.3 Preventing a STORE Statement from Taking Effect	5-3
5.4 Creating a New Record Without Complete Information	5-3
5.5 Storing Two or More Records At a Time	5-3
5.6 Storing the Correct Type of Data in New Records	5-4

6 Retrieving Data

6.1	Displaying All Records in a Domain	6-1
6.2	Selecting a Group of Records	6-2
6.2.1	Specifying the Number of Records in a Record Stream	6-2
6.2.2	Limiting Record Streams by Specifying Values	6-3
6.3	Examples of Displaying Data with the PRINT Statement	6-5
6.4	Sorting Records in Record Streams	6-7
6.5	Displaying Fields from Record Streams	6-9
6.6	Stopping the Display of Records	6-10
6.7	Using Collections	6-11
6.7.1	Forming Collections	6-11
6.7.2	Displaying the CURRENT Collection	6-11
6.7.3	Displaying Sorted Records in the CURRENT Collection	6-13
6.7.4	Displaying Information About the CURRENT Collection.	6-14
6.7.5	Making a New CURRENT Collection from an Old One	6-14
6.7.6	Using the CURRENT Collection in RSEs	6-15
6.8	Naming a Collection.	6-15
6.8.1	Showing the Names of Existing Collections	6-15
6.8.2	Naming an Unnamed Collection.	6-18
6.8.3	Changing the Content of a Named Collection	6-19
6.9	Releasing Collections	6-20
6.10	Using the FINISH Command	6-21

7 Modifying Data

7.1	Using the SELECT Statement with a Collection	7-1
7.2	Using Other Forms of the SELECT Statement.	7-5
7.3	Using the SHOW Command to Identify the Selected Record	7-7
7.4	Modifying a Selected Record in a Collection	7-7
7.5	Modifying Fields in Indexed Data Files	7-9
7.6	Modifying All Records in a Collection	7-9
7.7	Modifying Records in a Record Stream.	7-9
7.8	Modifying Specific Fields in a Record Stream	7-10

8 Erasing Data

8.1	Using the ERASE Statement on Indexed Files.	8-
8.1.1	Erasing a Selected Record	8-
8.1.2	Erasing a Collection of Records	8-
8.1.3	Erasing a Record Stream	8-
8.2	Using the MODIFY Statement on Sequential Files	8-

9 Writing Simple Reports

9.1	The General Features of a Report	9-1
9.2	The Report Specification.	9-2
9.2.1	Multiple-Line Report Names.	9-3
9.2.2	The Columns_Page Option	9-3
9.2.3	Print List Options	9-5
9.3	Output of Reports	9-5
9.3.1	Output to a Line Printer	9-5
9.3.2	Output to a Disk File	9-5
9.4	Saving Report Specifications.	9-6
9.5	Summary	9-6

Glossary

Index

Figures

1-1	The DATATRIEVE Domain	1-3
4-1	The HELP Screen	4-1
4-2	The HELP SHOW screen	4-2

Tables

6-1	Relational Operators	6-4
-----	--------------------------------	-----

How to Use This Manual

This book introduces you to DATATRIEVE–11. It begins with a brief explanation of information management and the use of DATATRIEVE. It then guides you through the basic tasks you can perform with DATATRIEVE.

Intended Audience

The *Introduction to DATATRIEVE–11* assumes you can use your terminal to log in to your local computer system and display the contents of your file directory. If you need to review how to do that, consult the user's guide for your operating system.

This book addresses people who are unfamiliar with DATATRIEVE–11.

Structure

There are nine chapters in this introduction:

- Chapter 1 Briefly describes information management and lists some basic DATATRIEVE functions
- Chapter 2 Supplies information about what you have to do to begin using DATATRIEVE, describes some of the basic commands and statements that tell DATATRIEVE what to do, and tells you how to leave DATATRIEVE
- Chapter 3 Explains how to define information for DATATRIEVE using ADT
- Chapter 4 Describes the use of Help and Guide Mode to give you online assistance while using DATATRIEVE
- Chapter 5 Explains how to store information with DATATRIEVE
- Chapter 6 Describes a variety of ways you can retrieve data

- Chapter 7 Contains information about modifying information you have stored previously
- Chapter 8 Explains how to erase data
- Chapter 9 Describes some of the basic features of the DATATRIEVE Report Writer

Conventions

The following list explains the special symbols used in this book:

RET

This symbol tells you to press the RETURN key on the keyboard of your terminal.

TAB

This symbol tells you to press the TAB key on the keyboard of your terminal.

CTRL/X

This symbol tells you to press the CTRL (control) key and a letter key (usually Z or C) at the same time.

>

The symbol > on a line by itself represents the system prompt.

color

Text printed in red shows the lines you type while doing the examples.

Related Manuals

DATATRIEVE-11 Guide to Writing Reports

DATATRIEVE-11 User's Guide

Introduction **1**

DATATRIEVE-11 is a tool for managing information. Its commands and statements, the words and phrases that make it run, are common English words. This book explains enough of the rules of DATATRIEVE for you to begin to use it.

Information management is a shorthand term for a process that consists of the following tasks:

- Setting up some related information (the data you are managing)
- Entering information
- Retrieving information
- Adding new information
- Deleting information
- Changing information

Some of the terminology may be strange to you, but the process itself should be familiar. If you have an address book, for instance, you manage information by:

- Picking the style of address book that best meets your needs
- Writing the original names and addresses in the book
- Finding the address of a restaurant to try
- Adding your new dentist's name and address
- Deleting the entry for your previous dentist
- Changing a friend's address when the family moves to another city

DATATRIEVE allows you to manage large amounts of information in complex ways, but the process of data management remains the same as for your address book.

1.1 DATATRIEVE Terminology

DATATRIEVE uses a vocabulary that may, in part, be unfamiliar to you. Terms you might find unfamiliar are explained when they first appear. With practice and experience, you will become familiar with these terms.

To help you, there is a short glossary at the end of this book. In addition, there is a more extensive glossary in the *DATATRIEVE-11 Summary Description*.

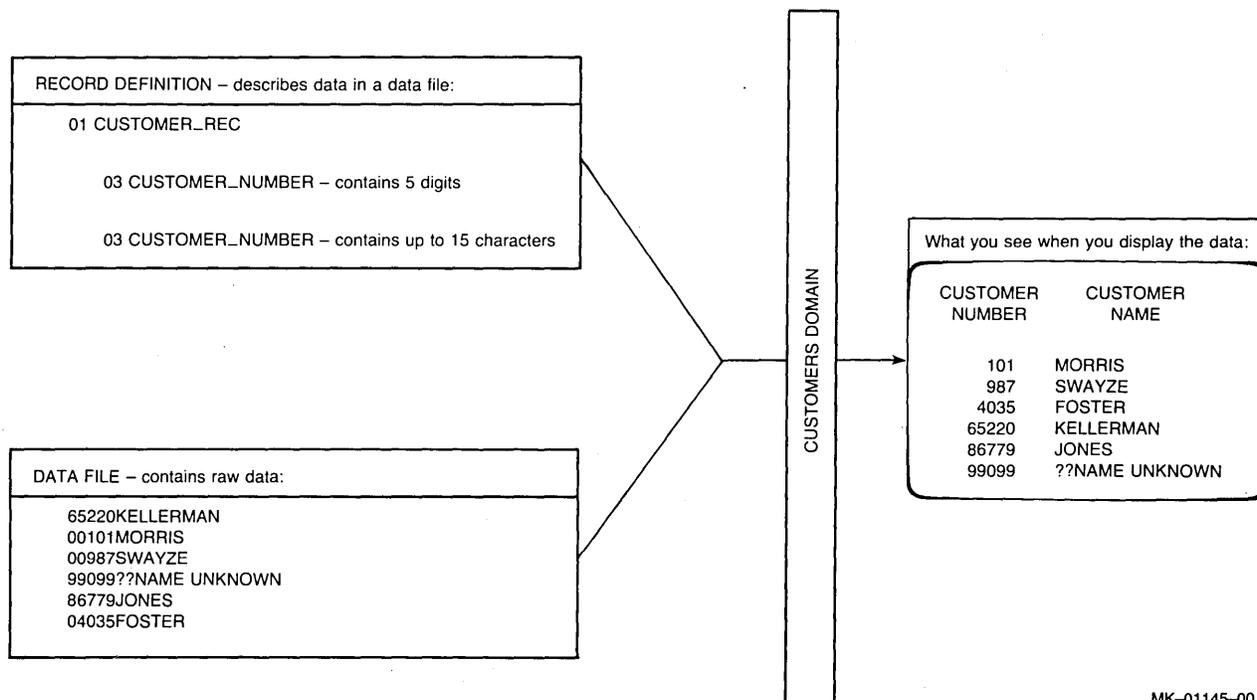
1.2 Using DATATRIEVE-11

This book is designed to help you use the basic commands and statements of DATATRIEVE. This introduction will teach you how to:

- Use the Application Design Tool (ADT) to define your own items of information
- Ready your data to gain access to it
- Print your data
- Select items of data that are related in a particular way
- Sort your data in various ways
- Edit errors in commands and statements
- Store new data
- Modify existing data
- Erase unneeded data
- Report your data

1.3 The Sample Domains

A record definition defines the organization of information contained in a data file, which is a collection of related records. The domain associates a data file with a record definition. The domain contains no data itself. When you use the domain name, you tell DATATRIEVE to use a particular record definition to interpret the data stored in a specific file. Figure 1-1 illustrates the relationships of the parts of a domain.



MK-01145-00

Figure 1-1: The DATATRIEVE Domain

When you practice using DATATRIEVE, you can use the four sample domains installed with DATATRIEVE:

- FAMILIES
- OWNERS
- PERSONNEL
- YACHTS

This book uses the PERSONNEL domain to show how to manage information with DATATRIEVE. You may use any of the domains supplied, or create your own domain so that you have a chance to work with data that is familiar to you. You may find it to be most helpful to practice with a domain you defined for yourself, using the examples in this book as guidelines.

If you use the PERSONNEL domain and do all the examples shown in this book, the database will change because you are making real changes when you follow the examples. If you do some but not all of the examples, some of your later results may differ from those printed in the book.

Starting and Stopping a DATATRIEVE Session **2**

This chapter shows you how to:

- Start DATATRIEVE
- Show what data is available to you
- Get ready to display your data
- Stop DATATRIEVE

2.1 Starting DATATRIEVE

To start DATATRIEVE, type:

```
DTR(RET)
```

DATATRIEVE displays a message resembling the following:

```
DATATRIEVE-11, DEC query and Report System  
Version: V3.00, 10-OCT-83  
Type HELP for help  
DTR>
```

If you do not get this startup banner, type either `RUN $DTR` or the symbol defined by your system manager to invoke DATATRIEVE.

The examples in this book use the sample domains `YACHTS`, `PERSONNEL`, `OWNERS`, and `FAMILIES`. The first time you use DATATRIEVE you should use the following command to create your own DATATRIEVE dictionary and copy the data files for `YACHTS`, `PERSONNEL`, `OWNERS`, and `FAMILIES` into your default directory. This ensures that your display of the examples will match the results printed in the book.

For RSTS/E systems type:

```
DTR> @LB:SETUP.DTR(RET)
```

For RSX systems type:

```
DTR> @LB:[1,2]SETUP.DTR(RET)
```

This command creates a dictionary for you called SAMPLE.DIC, enters the domain and record definitions into your dictionary, and copies the data files into your directory. The command displays the following information:

```
*****  
Datatrieve is now setting up its sample definitions and data in  
your private dictionary SAMPLE.DIC. The data files will be stored  
in your default directory.  
  
There will be 4 data files created:  
YACHT, OWNER, FAMILY and PERSON.  
*****
```

The commands and definitions scroll on your screen, and the display concludes with the message:

```
*****  
The Datatrieve sample definitions and data have been installed.  
*****  
DTR>
```

If you do not know whether you are using a RSTS/E or a RSX system, ask your system manager.

In response to the @SETUP.DTR command, DATATRIEVE may display a message that SAMPLE.DIC already exists. You can ignore this message.

To display the name of your dictionary, type:

```
DTR> SHOW DICTIONARY(RET)
```

For RSTS/E systems DATATRIEVE responds with a message resembling the following:

```
The current dictionary is SY:[1,170]SAMPLE.DIC  
DTR>
```

The message for RSX systems resembles the following:

```
The current dictionary is DB0:[1,2]SAMPLE.DIC  
DTR>
```

You can easily access and display information about your dictionary and its contents whenever you enter DATATRIEVE by executing a command file containing DATATRIEVE statements to make SAMPLE.DIC your default dictionary and display its contents. Follow these steps:

1. Exit from DATATRIEVE by typing:

```
DTR> EXIT(RET)
```

2. Create a command file called SAMPLE.CMD and enter the two statements:

```
SET DICTIONARY SAMPLE.DIC  
SHOW ALL
```

Now start DATATRIEVE by typing DTR at the system prompt just as you did at the beginning of this chapter. When you see the DTR> prompt again, execute the command file by typing @SAMPLE.CMD. On your terminal DATATRIEVE displays information resembling the following:

```
DTR> @SAMPLE.CMD(RET)  
SET DICTIONARY SAMPLE.DIC  
SHOW ALL  
Domains:  
    FAMILIES          KETCHES          OLD_FAMILIES     OWNERS  
    OWNERS_SEQUENTIAL PERSONNEL         PERSONNEL_SEQ  
    SAILBOATS         YACHTS           YACHTS_SEQUENTIAL  
Records:  
    FAMILY_REC       OWNER_RECORD     PERSONNEL_REC    PERSONNEL_SEQ_REC  
    YACHT  
Procedures:  
Tables:  
    RIG_TABLE  
The current dictionary is DB0:[100,170]SAMPLE.DIC  
No established collections  
No ready domains  
DTR>
```



2.2 Displaying Information About Your DATATRIEVE Environment

To display information about your DATATRIEVE environment, you use variations of the SHOW command. For example, in the command file you just created, the SHOW ALL command displays the entire contents of your dictionary.

You can also display parts of your dictionary contents by using other forms of the SHOW command. To display just the domains you have available, for example, enter:

```
DTR> SHOW DOMAINS(RET)
Domains:
          FAMILIES          KETCHES          OLD_FAMILIES          OWNERS
          OWNERS_SEQUENTIAL  PERSONNEL      PERSONNEL_SEQ
          SAILBOATS          YACHTS          YACHTS_SEQUENTIAL
DTR>
```

You can also use the SHOW RECORDS command to produce a display of records available to you. The domains, records, and other categories of information about your DATATRIEVE environment that you can display with the SHOW command are called objects. Other forms of the SHOW command are discussed later in this book.

2.3 Accessing Domains

To begin working with DATATRIEVE, you must first have access to the domain that you want to use. Preparing the domain for access is known as readying the domain.

To ready a domain, type READY followed by the name of the domain. For instance, to ready the PERSONNEL domain, type:

```
DTR> READY PERSONNEL (RET)
DTR>
```

The DTR> prompt tells you that the PERSONNEL domain is now available to you and that DATATRIEVE is waiting for your next command.

You can ready two or more domains at any one time:

```
DTR> READY YACHTS(RET)
DTR> READY FAMILIES(RET)
DTR>
```

To see what domains you have readied, use the SHOW READY command:

```
DTR> SHOW READY(RET)
Ready domains:
          FAMILIES: RMS SEQUENTIAL, PROTECTED READ
          YACHTS: RMS INDEXED, PROTECTED READ
          PERSONNEL: RMS INDEXED, PROTECTED READ
DTR>
```

Do not be concerned at present with the phrase "RMS INDEXED" and "PROTECTED READ". Chapter 3 explains indexed files, and the *DATATRIEVE-11 Reference Manual* discusses "PROTECTED READ" and other access modes.

If you try to ready a domain that is not defined in your dictionary, DATATRIEVE sends you an error message. For example, type:

```
DTR> READY INVENTORY (RET)
"INVENTORY" has not been defined in the dictionary
Execution failed
DTR>
```

In the next chapter you will learn more about defining domains.

2.4 Ending a DATATRIEVE Session

When you see the DTR> prompt, you can exit from DATATRIEVE in two ways:

- Type EXIT and press RETURN

```
DTR> EXIT (RET)
>
```

- Press the CTRL and Z keys at the same time. You can use CTRL/Z at any place in the line before you press RETURN. Displays of two examples follow:

```
DTR> ^Z
>
```

```
DTR> SHOW REA^Z
>
```


Defining Data With ADT **3**

This book uses examples from the PERSONNEL domain to teach you some of the uses of DATATRIEVE. The PERSONNEL domain already exists. However, when you use DATATRIEVE in your work, you create your own domains.

The Application Design Tool (ADT) provides a fast way to define a domain and the record definition and data file that go with the domain. ADT prompts you for all necessary information and sends you messages explaining your choices.

During a session, ADT prompts for:

- The domain name
- The data file name
- The name of each field
- The kind of information in each field, for example characters, numbers, dates, or money
- The format of fields containing dates, numbers, or money
- The length of fields containing characters or numbers

At the end of the ADT session, you can begin to store your data. The Application Design Tool is especially helpful when you are beginning to use DATATRIEVE and want to define your own domains. As you are gaining experience, ADT can help you create the basic definitions. You can later edit these basic definitions to include other features.

3.1 Planning a DATATRIEVE Domain

Before invoking the Application Design Tool, decide what information you want to store. For example, if you are interested in doing some comparison shopping before you buy a new car, you might want to store the following information about each car:

- The manufacturer
- The model of the car
- The model year
- The date you visited the show room
- The price
- The options available

You can consider each piece of information as a field in a record. The order in which you enter the fields determines the order DATATRIEVE normally displays the fields on your terminal or prints them in a report.

Next, decide the characteristics of each field:

- Name

You can choose any name you want for each field as long as the name begins with a letter and does not exceed 31 characters. Possible field names are MAKER, MODEL, YEAR, VISIT_DATE, PRICE and OPTIONS.

- Query Name

In DATATRIEVE you can use query names as short and easy substitutes for long field names. A query name can replace the complete field name in any DATATRIEVE statement. ADT asks you to supply a query name for any field longer than seven characters. For example, ADT asks you for a query name for the field name VISIT_DATE because VISIT_DATE is longer than seven characters. You could supply the query name VISIT for this field. If you do not want to use a query name, press RETURN and continue using ADT.

- Data Type

Decide what kind of information each field contains:

- Character: as many letters, numbers, and other characters as you specify
- Date: a choice of four formats
- Number: as many digits as you specify, with or without a decimal point
- Money: a number preceded by a dollar sign (\$) and followed by a decimal point and two digits
- Percentage: a number with a percent sign (%)

The MAKER, MODEL, and OPTIONS fields contain character data. The YEAR field contains a number. The VISIT_DATE field holds date data, and the PRICE field consists of money.

- Size

The number of letters, numbers, or other characters you need for each field determines the size of the field. For example, the field for options might need 29 characters while the model year only needs two. The number of digits to the left of the decimal determines the size of a field containing money data. Make each field large enough to store the information you want, but not so large that the field wastes memory space.

At the beginning of an ADT session, ADT prompts you for the domain name. Choose a name that describes the information you are storing so that the name is meaningful and easy to remember. Since your data is stored on a file, ADT asks you for the name of the file. Again, choose a name that reflects the information you are storing. For example, in the ADT session for comparing cars, you could name the domain AUTOMOBILES and the file AUTO.DAT.

At the end of the ADT session, you choose:

- The type of data file (sequential or indexed)
- The name and attributes of each index key for an indexed file
- The name of a command file to contain the definitions

A comparison of sequential and indexed files follows:

Sequential Files	Indexed Files
Records are stored in the order you enter them.	Records are stored according to the order of the primary field (alphabetically for a field containing characters and numerically for a field containing numbers).
Access to information is slow.	Access to information is fast.
You can change data in any field.	You can change data in any field except the primary key field.
You cannot erase records from the file.	You can erase records from the file.

Generally, it is useful to specify an indexed file.

If you want an indexed file, ADT asks you for the name of the primary key — the field that determines the order for storing the records in the file. If you want your automobile record in alphabetical order by maker, for example, you should specify the MAKER field as the primary key.

Once you have named the primary key, ADT asks you to decide whether you want to allow duplicate values for that field. If MAKER is your primary key, you

need to allow duplicate values, since one maker (Chevrolet, for instance) has many models of cars.

You cannot change the information in a primary key.

After you store data for **MAKER**, you cannot change it unless you erase the whole record and store a new one in its place. In other words, you can have as many Chevrolets as you like listed as makers, but you can never change a Chevrolet to a Ford.

ADT also allows you to name alternate index keys. If you plan to perform queries frequently that involve another field, you can name that field as an alternate index key when ADT prompts you for an alternate index key.

3.2 Using ADT

The next two sections of this chapter illustrate an ADT session that stores information about cars. A summary of the session follows:

- The domain-name: AUTOMOBILES
- The file name for the data: AUTO.DAT
- The car record fields:

Field Name	Query Name	Data Type	Field Size
MAKER	None	Character	10
MODEL	None	Character	10
YEAR	None	Number	2
VISIT_DATE	VISIT	Date	
PRICE	None	Money	5
OPTIONS	None	Character	29

- File type: Indexed
- Field name of the primary key: MAKER
- Duplicates for the primary field key: Yes
- Alternate index keys: No
- The file name for the domain and record definitions: AUTOS

You can create this domain yourself by following the example. The most helpful thing you could do for your understanding of the process, however, is to create a domain of your own. Use some type of data familiar to you, and think of the types of reports and queries you want to create using this data.

To start an ADT session, type:

```
DTR> ADT(RET)
```

The following session shows the questions that appear on your screen with sample answers. Responses to the first entries are question marks to show you the kind of information that ADT provides when you ask for help. Use the question mark yourself as often as you need:

```
DTR> ADT(RET)
```

```
Do you want help? (YES or NO) : Y(RET)
```

```
ADT will help you create your DATATRIEVE domain and record definitions. The DTR statements will be written to the file you specify.
```

```
All answers must be terminated by a carriage return.
```

```
If you need help for any question, type ? followed by a carriage return.
```

```
If you would like to see what you have defined, enter ! followed by a carriage return.
```

```
Do you want detailed questions? (YES or NO) : ?(RET)
```

```
Detailed questions are available if you are not familiar with ADT.
```

```
Abbreviated questions assume that you already know ADT concepts and terminology.
```

```
Help is available for both types of questions.
```

```
Do you want detailed questions? (YES or NO) : YES(RET)
```

```
What do you want to name this domain? : ?(RET)
```

```
A domain is a logically related collection of data which is treated as a unit by DATATRIEVE. You can also think of a domain as a file.
```

```
What do you want to name this domain? : AUTOMOBILES(RET)
```

```
What do you want to name the file where the data for AUTOMOBILES will be? : AUTO.DAT(RET)
```

```
What do you want to name the first field in AUTOMOBILES? : MAKER(RET)
```

```
What's in MAKER --
```

```
  a DATE
```

```
  a PERCENT
```

```
  MONEY
```

```
  NUMBERS used in arithmetic
```

```
  or ANYTHING ELSE (CHARACTERS)?
```

```
Enter one of the above : CHARACTERS(RET)
```

```
How many characters long is MAKER ? : 10(RET)
```

```
Are there any more fields in AUTOMOBILES ? (YES or NO) : YES(RET)
```

```
What do you want to name the next field in AUTOMOBILES? : MODEL(RET)
```

```
What's in MODEL --
```

```
  a DATE
```

```
  a PERCENT
```

```
  MONEY
```

```
  NUMBERS used in arithmetic
```

```
  or ANYTHING ELSE (CHARACTERS)?
```

You can specify a field's data type quickly and easily. Simply enter the first letter of the data type you choose.

Similarly, you can type either a Y or N in response to questions requiring a yes or no answer.

```
Enter one of the above : C(RET)
How many characters long is MODEL ? : 10(RET)
Are there any more fields in AUTOMOBILES ? (YES or NO) : Y(RET)
What do you want to name the next field in AUTOMOBILES? : YEAR(RET)
What's in YEAR --
  a DATE
  a PERCENT
  MONEY
  NUMBERS used in arithmetic
  or ANYTHING ELSE (CHARACTERS)?
```

```
Enter one of the above : NUMBERS(RET)
How many digits to the left of the decimal point? : 2(RET)
How many digits to the right of the decimal point? : 0(RET)
Do you want zeros to the left to print as spaces? : ?(RET)
If you do not suppress zeros to the left of the number, they
will be printed (ex. zip code = 00274 instead of 274).
Do you want zeros to the left to print as spaces? Y(RET)
Are there any more fields in AUTOMOBILES ? (YES or NO) : YES(RET)
What do you want to name the next field in AUTOMOBILES? : VISIT_DATE(RET)
What is the query abbreviation for VISIT_DATE? : VISIT(RET)
What's in VISIT_DATE --
  a DATE
  a PERCENT
  MONEY
  NUMBERS used in arithmetic
  or ANYTHING ELSE (CHARACTERS)?
Enter one of the above : DATE(RET)
```

ADT prompts you for one of four date formats. You can choose whichever format you prefer.

```
Four date formats are available:
  1      MM/DD/YY      6/29/79
  2      DD-MMM-YY     29-JUN-79
  3      DD-MMM-YYYY   29-JUN-1979
  4      DD.MM.YY     29.06.79
Enter format number 1, 2, 3, or 4 : 3(RET)
Are there any more fields in AUTOMOBILES ? (YES or NO) : YES(RET)
What do you want to name the next field in AUTOMOBILES? : PRICE(RET)
What's in PRICE --
  a DATE
  a PERCENT
  MONEY
  NUMBERS used in arithmetic
  or ANYTHING ELSE (CHARACTERS)?
```

```
Enter one of the above : MONEY(RET)
How many digits to the left of the decimal point? : 5(RET)
```

ADT formats a money field in this way:

- A \$ sign before the amount
- A decimal point
- Either two 0s after the decimal point if you type only a dollar amount (for example, 6755) or the cents if you type a dollar and cents amount (for example, 6755.98)

```
Are there any more fields in AUTOMOBILES ? (YES or NO) : YES(RET)
What do you want to name the next field in AUTOMOBILES? : OPTIONS(RET)
What's in OPTIONS --
    a DATE
    a PERCENT
    MONEY
    NUMBERS used in arithmetic
    or ANYTHING ELSE (CHARACTERS)?
```

```
Enter one of the above : CHARACTERS(RET)
How many characters long is OPTIONS ? : 29(RET)
Are there any more fields in AUTOMOBILES ? (YES or NO) : NO(RET)
An indexed file can handle certain queries based on a key
field very quickly. A sequential file is not as fast and
does not allow records to be ERASEd. BUT, an indexed file
does not allow you to change the primary key field's data.
Do you want your data file to be indexed? (YES or NO) : YES(RET)
What is the field name of the primary key? : MAKER(RET)
Do you want to allow the field which is the primary key
to have duplicates? (YES or NO) : YES(RET)
Do you want alternate keys? (YES or NO) NO(RET)
What is the name of the file where the DATATRIEVE
domain and field definitions should go? : AUTOS(RET)
The DATATRIEVE definitions for your domain are
located in file AUTOS.CMD;
The record length is 66 bytes.
Do you want to define another domain? (YES or NO) : NO(RET)
DTR>
```

3.2.1 Invoking the .CMD File

If you completed the above steps, a new file, AUTOS.CMD is placed in your directory. This file contains the definitions of the domain, record, and data file you created during your ADT session.

You must enter these definitions into your dictionary so you can use them in a DATATRIEVE session. To insert the definitions into your dictionary type @AUTOS. The definitions in AUTOS.COMD scroll on the screen:

```
DTR>@AUTOS(RET)
DEFINE DOMAIN AUTOMOBILES USING AUTOMOBILES-REC
      ON AUTO.DAT;
DEFINE RECORD AUTOMOBILES-REC USING
01 AUTOMOBILES-REC,
  15 MAKER      PIC IS X(10),
  15 MODEL      PIC IS X(10),
  15 YEAR       PIC IS S9(2)  EDIT-STRING IS -Z(2),
  15 VISIT_DATE  USAGE IS DATE
      EDIT_STRING IS DD-MMM-YYYY
      QUERY_NAME IS VISIT,
  15 PRICE      PIC IS S9(5)V99
      EDIT_STRING IS $$$,$$$,$$,
  15 OPTIONS    PIC IS X(29),
;
[Record AUTOMOBILES_REC is 66 bytes long]
DEFINE FILE FOR AUTOMOBILES      KEY=MAKER (DUP);
DTR>
```

DATATRIEVE inserts the definitions into your current dictionary and creates a data file, AUTO.DAT, in your default directory.

When you define your own domains, records, and files, you can place the definitions in separate dictionaries. To create a new dictionary, type:

```
DEFINE DICTIONARY dictionary-name
```

To place the definitions in this dictionary:

1. Type SET DICTIONARY dictionary-name
2. Invoke the command file containing the record definitions

3.3 Defining Domains Outside ADT

In this chapter you defined the domain AUTOMOBILES, the record AUTOMOBILES_REC, and the file AUTO.DAT using ADT. You provided the information, and ADT created the definitions for you. You can define domains, records, and files on your own using these commands:

- DEFINE DOMAIN
- DEFINE RECORD
- DEFINE FILE

These commands give you a number of options not available in ADT. The record for the PERSONNEL domain, used throughout this book, contains some of these additional features. The record definition PERSONNEL_REC is shown below:

```
DTR> SHOW PERSONNEL_REC (RET)
RECORD PERSONNEL_REC
  USING
01 PERSON.
   05 ID                PIC IS 9(5).
   05 EMPLOYEE_STATUS  PIC IS X(11)
                        QUERY_NAME IS STATUS
                        QUERY_HEADER IS "STATUS"
                        VALID IF STATUS EQ "TRAINEE","EXPERIENCED".
   05 EMPLOYEE_NAME
     10 FIRST_NAME     PIC IS X(10)
                        QUERY_NAME IS F_NAME.
     10 LAST_NAME      PIC IS X(10)
                        QUERY_NAME IS L_NAME.
   05 DEPT              PIC IS XXX.
   05 START_DATE       USAGE IS DATE.
   05 SALARY           PIC IS 9(5)
                        EDIT_STRING IS $$$,$$$,.
   05 SUP_ID           PIC IS 9(5).
```

As you use this book, you will learn more about this record definition.

This book assumes you will use the sample domains or create your own domains with ADT. If at this point you want to use the DEFINE command to create your own domains, you can find further information in the *DATATRIEVE-11 User's Guide*. You should also know about one or more of the techniques that can help you avoid retyping a long series of DATATRIEVE statements, either because you made an error or because you want to use the statements again. These techniques are command procedures, DATATRIEVE procedures, the DATATRIEVE editor, and EXTRACT. These are also explained in the *DATATRIEVE-11 User's Guide*.



MK-01109-00

HELP and GUIDE Mode: Quick-Reference Section **4**

4.1 Using DATATRIEVE HELP

A HELP program is available that can answer many questions you may have as you are using DATATRIEVE. You can call it by typing HELP and pressing RETURN whenever you are at the DTR> prompt.

When you request HELP, your screen appears as shown in Figure 4-1.

```
DTR> HELP(RET)
Two levels of assistance are available, basic and advanced.
To get help for the elementary DATATRIEVE statements, type
HELP followed by the names of the statements for which help
is required. To get a list of the topics for which help is
available, type
```

```
HELP HELP
```

```
To get assistance with advanced DATATRIEVE statements, type
```

```
HELP ADVANCED
```

```
followed by the names of the statements for which help is
required. To get a list of the topics for which advanced
help is available, type
```

```
HELP ADVANCED HELP
```

```
Paired square brackets indicate that the enclosed clause is
optional. Capital letters indicate DATATRIEVE keywords.
Words in lower case indicate that a user supplied name or
value is required.
```

```
DTR>
```

Figure 4-1: The HELP Screen

After you become familiar with the HELP program, just type HELP and the name of the topic you are interested in, and the appropriate text appears on the screen. Probably the best way to become familiar with HELP is to type HELP and then simply browse around among the subjects available. In the process, you can also learn about the DATATRIEVE topics you read.

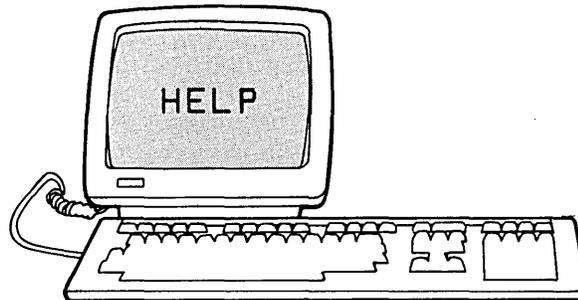
Figure 4-2 shows HELP for a sample subject.

```
DTR> HELP SHOW(RET)
The SHOW statement is used to display the definitions of
domains, records, procedures, and tables stored in the data
dictionary, and information concerning the state of various
objects within Datatrieve. The form of the statement is:

        SHOW object-1, object-2, object-3,...

where each "object" is one of the followings:
    SETS
    DOMAINS
    RECORDS
    PROCEDURES
    TABLES
    COLLECTIONS
    FIELDS
    READY
    CURRENT
    ALL
    DICTIONARY
    SPACE
    domain-name
    record-name
    procedure-name
    table-name
    collection-name
DTR>
```

Figure 4-2: The HELP SHOW Screen



MK-01101-00

4.2 Using GUIDE Mode

DATATRIEVE offers another type of help in GUIDE mode. GUIDE mode is useful when you are just learning DATATRIEVE, because it prompts you with a choice of entries that are correct.

GUIDE mode works only on DIGITAL's VT family of terminals. If you have a hard-copy terminal and enter a request for GUIDE mode, DATATRIEVE displays an error message on your terminal and returns you to DATATRIEVE command level (indicated by the DTR> prompt).

To invoke GUIDE mode, enter:

```
DTR> SET GUIDE(RET)
```

To see how GUIDE mode works, you must use it. You can use GUIDE mode to help you make entries; it completes the typing of keywords if you press the space bar after it has enough letters to uniquely identify the word you are typing.

If READY is the only possible entry at a certain point, for example, you can type R and press the space bar. Guide mode completes the word for you. If you type "RE," "REA," and so on, it also completes the word as READY.

Occasionally the second letter of the command duplicates the first letter of your next word. Suppose you want to ready a domain called EMPLOYEES. If you type "R" "E," GUIDE mode sees the "E" as the second letter in READY. You must type "RE" and then the "E" of EMPLOYEES.

When you enter SET GUIDE and press RETURN, DATATRIEVE clears the terminal screen and prompts you to enter a command or type a question mark (?) for help. If you request help, DATATRIEVE displays all the possible words you can currently use.

Throughout, GUIDE mode prompts you for the next part of your entry, including spaces, quotation marks, and carriage returns. Not all options are acceptable in GUIDE mode, but you can find out which ones are at any time by typing a question mark (?), and DATATRIEVE displays all the acceptable alternatives. DATATRIEVE displays an error message if the entry you try is not allowed.

You can stop GUIDE mode by typing LEAVE. DATATRIEVE then displays a message on your terminal and returns you to DATATRIEVE command level (indicated by the DTR> prompt):

```
LEAVE(RET)
```

```
You're on your own now. Good luck!
```

```
DTR>
```

Although GUIDE mode is primarily a training tool, you may also find it useful to save keystrokes while typing. Experiment with it, and see the results you get. You can then judge how to use GUIDE mode to your advantage.

Storing Data **5**

This chapter shows you how to add new records to a data file using the STORE statement.

5.1 Preparing a Domain for Storing Data

To store new records in a data file, you must ready the domain for WRITE or EXTEND access. With the WRITE command you can store, read, erase, and change records. With the EXTEND command you can only store records. The examples in this chapter use the PERSONNEL domain readied for WRITE access:

```
DTR> READY PERSONNEL WRITE(RET)  
DTR>
```

The DTR> prompt indicates that DATATRIEVE has readied the domain. Verify the type of access by entering:

```
DTR> SHOW READY(RET)  
Ready domains:  
  
PERSONNEL: RMS INDEXED, PROTECTED WRITE  
  
DTR>
```

The phrase PROTECTED WRITE confirms that you have write access to PERSONNEL. You can now store records with the STORE statement.

5.2 Using the STORE Statement

To add new records to a data file, use the STORE statement. DATATRIEVE immediately prompts you for the value of each elementary field in the new record. To store a new record in the PERSONNEL domain, type:

```
DTR> STORE PERSONNEL(RET)
Enter ID: 87422(RET)
Enter EMPLOYEE_STATUS: EXPERIENCED(RET)
Enter FIRST_NAME: MARTY(RET)
Enter LAST_NAME: KELLEY(RET)
Enter DEPT: T32(RET)
Enter START_DATE: 26-AUG-1982(RET)
Enter SALARY: 18750(RET)
Enter SUP_ID: 87289(RET)
DTR>
```

Storing a record gives you an opportunity to observe several DATATRIEVE features you have not encountered before.

Since in certain circumstances DATATRIEVE is case sensitive, you should make a habit of using all uppercase letters.

Note that DATATRIEVE displays a date in the format you specify in the record definition, not in the format you use when you store the date. In the example above, you entered the date as 26-AUG-1982. You could also have entered it as 082682. In either case, DATATRIEVE displays the date as 26-Aug-1982.

Similarly, the record definition determines the format for the SALARY field. You enter the salary without a dollar sign. The record definition determines the form the field takes as output. The PERSONNEL_REC specifies that the dollar sign be printed.

If you did the preceding example using your own PERSONNEL database, you actually added this employee to your file.



MK-01105-00

5.3 Preventing a STORE Statement from Taking Effect

To prevent the STORE statement from creating a new record, type CTRL/Z at any point before you press RETURN to enter a value in the last field of the record:

```
DTR> STORE PERSONNEL(RET)
Enter ID: 07391(RET)
Enter EMPLOYEE_STATUS: TRAINEE (RET)
Enter FIRST_NAME: CHARLEY(RET)
Enter LAST_NAME: ^Z
Execution terminated by operator
DTR>
```

5.4 Creating a New Record Without Complete Information

You can create a new record, even though all the necessary information is not yet available to you. In the following example, a person has been hired but her starting date, salary, and assignment are unknown. If you enter a SPACE or TAB before pressing RETURN, DATATRIEVE sets the field to blank or zero, depending on the field definition. If you press RETURN without making any entry, DATATRIEVE continues to prompt you until you enter something.

```
DTR> STORE PERSONNEL(RET)
Enter ID: (RET)
Enter ID: 12345(RET)
Enter EMPLOYEE_STATUS: TRAINEE(RET)
Enter FIRST_NAME: HENRIETTA(RET)
Enter LAST_NAME: MILLER(RET)
Enter DEPT: (TAB)(RET)
Enter START_DATE: (TAB)(RET)
Enter SALARY: (TAB)(RET)
Enter SUP_ID: (TAB)(RET)

DTR>
```

If you did this example, the entry is now included in your PERSONNEL database, with blanks in the DEPT, START_DATE, and SALARY fields, and 00000 in the SUP_ID field. The PERSONNEL_REC record definition sets all these defaults.

5.5 Storing Two or More Records At a Time

You can store more than one record at a time by using the REPEAT statement. Type REPEAT, the number of records you want to STORE, and the statement STORE PERSONNEL. DATATRIEVE prompts you for the values of each elementary field in each new record:

```
DTR> REPEAT 2 STORE PERSONNEL(RET)
Enter ID: 75639(RET)
Enter EMPLOYEE_STATUS: TRAINEE(RET)
Enter FIRST_NAME: MIKE(RET)
Enter LAST_NAME: MILLER(RET)
Enter DEPT: T44(RET)
Enter START_DATE: 6-FEB-1981(RET)
```

(continued on next page)

```

Enter SALARY: 16750(RET)
Enter SUP_ID: 00012(RET)
Enter ID: 75640(RET)
Enter STATUS: TRAINEE(RET)
Enter FIRST_NAME: BEN(RET)
Enter LAST_NAME: PUTNAM(RET)
Enter DEPT: T44(RET)
Enter START_DATE: 6-FEB-1981(RET)
Enter SALARY: 16500(RET)
Enter SUP_ID: 00012(RET)
DTR>

```

Again, if you did this example, you have added these two employees to your PERSONNEL database.

When you are storing more than one record with a STORE command, each single record is stored as you complete the information for it and press the last RETURN before continuing with the next record. If you have requested DATATRIEVE to store a certain number of records and it turns out that you want to enter fewer than you had planned, you can stop the prompting for more entries with CTRL/Z.

If you use CTRL/Z to terminate the entry of a record, it affects only that record.

5.6 Storing the Correct Type of Data in New Records

The record definition specifies the type of data that you can store in each field of the record. When you create a new record, you must put the correct type of data in each of the fields. For instance, if the record definition requires a number, DATATRIEVE refuses to accept letters. If you try to enter the wrong type of data in a field, DATATRIEVE prompts you again for valid data.

Look at the PERSONNEL_REC displayed in section 3.3. To the right of each field name the record contains information describing that field. The PIC IS clause tells you the field's data type and its length. Xs indicate character data, and 9s indicate number data. The number in parentheses tells you the length of the field. For example, in the field SUP_ID, the PIC IS clause tells you that you can enter up to five numbers for SUP_ID. Sometimes the length of the field is described as a series of 9s or Xs, as in the DEPT field. The three Xs tell you that DEPT contains three characters.

A summary of the type of data for the PERSONNEL_REC follows:

- EMPLOYEE_STATUS, FIRST_NAME, LAST_NAME, and DEPT accept only character data.
- START_DATE accepts only a date.
- SALARY and SUP_ID accept only numbers.

Sometimes records contain VALID IF clauses. For example, notice the clause VALID IF STATUS EQ "TRAINEE", "EXPERIENCED" in the field EMPLOYEE_STATUS. EQ is an expression meaning equal. The VALID IF clause specifies the choices that you can use for a particular field. In the EMPLOYEE_STATUS field the VALID IF clause tells you to enter either

TRAINEE or **EXPERIENCED** with capital letters when storing information in that field. **DATATRIEVE** does not accept any other entry for the field **EMPLOYEE_STATUS**.

The next example shows **DATATRIEVE**'s response to several of these kinds of entries:

```
DTR> STORE PERSONNEL(RET)
Enter ID: HAROLD(RET)
Non-digit in string "HAROLD", ignoring character(s)
Re-enter ID: 534020(RET)
Truncation during assignment
Re-enter ID: 34020(RET)
Enter EMPLOYEE_STATUS: BEGINNER(RET)
Validation error for EMPLOYEE_STATUS
Re-enter EMPLOYEE_STATUS: TYRO(RET)
Validation error for EMPLOYEE_STATUS
Re-enter EMPLOYEE_STATUS: TRAINEE(RET)
Enter FIRST_NAME: ^Z
Execution terminated by operator
DTR>
```

The next chapter explains how to use the **PRINT** statement to retrieve data you have stored.

Retrieving Data 6

After you have readied a domain, you can display the data at your terminal. With the PRINT statement, you form a stream of records for display. The record stream you form can be all the records in the domain, or a group of records you select by specifying certain characteristics.

6.1 Displaying All Records in a Domain

If the domain does not contain a large number of records, you may want to display all the records it contains. You need type only a READY command and a PRINT statement. The record stream displayed with the following PRINT statement contains all the records in the PERSONNEL domain, including the ones you stored in the previous chapter. (You can also type PRINT ALL PERSONNEL to form the same record stream.)

```
DTR> READY PERSONNEL(RET)  
DTR> PRINT PERSONNEL(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	00012
00891	EXPERIENCED	FRED	HOWL	F11	9-Apr-76	\$59,594	00012
02943	EXPERIENCED	CASS	TERRY	D98	2-Jan-80	\$29,908	39485
12345	TRAINEE	HENRIETTA	MILLER				00000
12643	TRAINEE	JEFF	TASHKENT	C82	4-Apr-81	\$32,918	87465
32432	TRAINEE	THOMAS	SCHWEIK	F11	7-Nov-81	\$26,723	00891
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	87289
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	00012
38465	EXPERIENCED	JOANNE	FREIBURG	E46	20-Feb-80	\$23,908	48475
39485	EXPERIENCED	DEE	TERRICK	D98	2-May-77	\$55,829	00012
48475	EXPERIENCED	GAIL	CASSIDY	E46	2-May-78	\$55,407	00012
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	87289
49001	EXPERIENCED	DAN	ROBERTS	C82	7-Jul-79	\$41,395	87465
49843	TRAINEE	BART	HAMMER	D98	4-Aug-81	\$26,392	39485
75639	TRAINEE	MIKE	MILLER	T44	6-Feb-81	\$16,750	00012
75640	TRAINEE	BEN	PUTNAM	T44	6-Feb-81	\$16,500	00012
78923	EXPERIENCED	LYDIA	HARRISON	F11	19-Jun-79	\$40,747	00891
83764	EXPERIENCED	JIM	MEADER	T32	4-Apr-80	\$41,029	87289

(continued on next page)

84375	EXPERIENCED	MARY	NALEVO	D98	3-Jan-76	\$56,847	3940
87289	EXPERIENCED	LOUISE	DEPALMA	G20	28-Feb-79	\$57,598	0000
87422	EXPERIENCED	MARTY	KELLEY	T32	26-Aug-82	\$18,750	8720
87465	EXPERIENCED	ANTHONY	IACOBONE	C82	2-Jan-73	\$58,462	0000
87701	TRAINEE	NATHANIEL	CHONTZ	F11	28-Jan-82	\$24,502	0080
88001	EXPERIENCED	DAVID	LITELLA	G20	11-Nov-80	\$34,933	8720
90342	EXPERIENCED	BRUNO	DONCHIKOV	C82	9-Aug-78	\$35,952	8740
91023	TRAINEE	STAN	WITTGEN	G20	23-Dec-81	\$25,023	8720
99029	EXPERIENCED	RANDY	PODERESIAN	C82	24-May-79	\$33,738	8740

DTR>

DATATRIEVE displays the records in order according to the employees' ID number, which is the primary key.

The following section explains limited record streams, which print some but not all of the records.

6.2 Selecting a Group of Records

Most often, you want to retrieve only certain records from the domain. DATATRIEVE allows you to select records by using a record selection expression (RSE). The RSE is a phrase that DATATRIEVE uses to form a group of records, called a record stream. When you typed PRINT PERSONNEL in the previous example, you used the simplest form of an RSE, an entire domain.

These are also examples of RSEs:

- PRINT FIRST 3 PERSONNEL
- PRINT PERSONNEL WITH START_DATE = "1-MAR-82"
- PRINT PERSONNEL WITH LAST_NAME = "HARRISON"
- PRINT PERSONNEL WITH SUP_ID = 87289

6.2.1 Specifying the Number of Records in a Record Stream

You can limit the number of records in a record stream with the FIRST clause. Suppose you do not want to see all the PERSONNEL records but would like to see the first three. You can limit the record stream with the restriction clause, specifying that you want to display only the FIRST 3 PERSONNEL:

```
DTR> READY PERSONNEL(RET)
DTR> PRINT FIRST 3 PERSONNEL(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SU ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	000
00891	EXPERIENCED	FRED	HOWL	F11	9-Apr-76	\$59,594	000
02943	EXPERIENCED	CASS	TERRY	D98	2-Jan-80	\$29,908	394

DTR>

6.2.2 Limiting Record Streams by Specifying Values

You can also limit a record stream by using a WITH clause. The phrase PERSONNEL WITH SUP_ID = 87289 is a record selection expression (RSE). This phrase forms the record stream by defining specific conditions that individual records must meet to be included. In the following example, the record stream includes only those employees whose supervisor has the ID 87289:

```
DTR> PRINT PERSONNEL WITH SUP_ID = 87289(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	87289
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	87289
83764	EXPERIENCED	JIM	MEADER	T32	4-Apr-80	\$41,029	87289
87422	EXPERIENCED	MARTY	KELLEY	T32	26-Aug-82	\$18,750	87289
88001	EXPERIENCED	DAVID	LITELLA	G20	11-Nov-80	\$34,933	87289
91023	TRAINEE	STAN	WITTGEN	G20	23-Dec-81	\$25,023	87289

```
DTR>
```

The following examples show three different types of RSE:

- Display the records of the first five trainees in the PERSONNEL domain. The RSE is FIRST 5 PERSONNEL WITH STATUS = "TRAINEE". DATATRIEVE searches through the PERSONNEL file until it has five records that fit the requirements of the RSE.

```
DTR> PRINT FIRST 5 PERSONNEL WITH STATUS = "TRAINEE"(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
12345	TRAINEE	HENRIETTA	MILLER				00000
12643	TRAINEE	JEFF	TASHKENT	C82	4-Apr-81	\$32,918	87465
32432	TRAINEE	THOMAS	SCHWEIK	F11	7-Nov-81	\$26,723	00891
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	87289
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	87289

```
DTR>
```

- Display the records of employees who started work after 1-Jan-1982. The RSE is PERSONNEL WITH START_DATE GREATER_THAN "1-JAN-1982." DATATRIEVE compares that date with the value of the START_DATE field in every record in the PERSONNEL domain. When it finds a record with a date later than 1-Jan-1982, that record becomes part of the record stream.

```
DTR> PRINT PERSONNEL WITH START_DATE GREATER_THAN "1-JAN-1982"(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	87289
87422	EXPERIENCED	MARTY	KELLEY	T32	26-Aug-82	\$18,750	87289
87701	TRAINEE	NATHANIEL	CHONTZ	F11	28-Jan-82	\$24,502	00891

```
DTR>
```

- Display the records of employees who earn more than \$20,000 but less than \$25,000. The RSE is PERSONNEL WITH SALARY BETWEEN 20000 AND 25000. DATATRIEVE again searches through the entire database, selecting all employees whose salaries fall within that range.

DTR> PRINT PERSONNEL WITH SALARY BETWEEN 20000 AND 25000(RET)

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
38465	EXPERIENCED	JOANNE	FREIBURG	E46	20-Feb-80	\$23,908	48475
87701	TRAINEE	NATHANIEL	CHONTZ	F11	28-Jan-82	\$24,502	00891

DTR>

In these examples, GREATER_THAN, =, and BETWEEN control the comparison of the value in a field with the value you supply. They are called relational operators.

DATATRIEVE recognizes several relational operators. Table 6-1 lists the ones you can use in DATATRIEVE record selection expressions and shows examples of their use. Notice that you do not need quotation marks around numeric fields that follow the relational operators, but you do need them around character fields and date fields.

Table 6-1: Using Relational Operators

Specify One of:	To Mean:	For Example:
EQUAL EQ =	Equal to	STATUS = "TRAINEE"
NOT_EQUAL NE	Not equal to	DEPT NE "T32"
GREATER_THAN GT >	Greater than	SALARY > 32500 DATE GT "1-Jan-1982"
GREATER_EQUAL GE	Greater than or equal to	ID GE 89999
LESS_THAN LT <	Less than	DATE LT "31-Dec-1981" SALARY LT 28000
LESS_EQUAL LE	Less than or equal to	SALARY LE 30000
BETWEEN x AND y BETWEEN x y BT x AND y BT x y	Between and including the two values specified	SALARY BT 20000 30000
CONTAINING CONT	Containing the specified value	DEPT CONTAINING "F" STATUS CONT "T"

After EQUAL, EQ, or =, you can enter a series of values, using a comma to separate each conditional value from the next one in the series:

```
DTR> PRINT PERSONNEL WITH SUP_ID = 48475, 39485(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
02943	EXPERIENCED	CASS	TERRY	D98	2-Jan-80	\$29,908	39485
38465	EXPERIENCED	JOANNE	FREIBURG	E46	20-Feb-80	\$23,908	48475
49843	TRAINEE	BART	HAMMER	D98	4-Aug-81	\$26,392	39485
84375	EXPERIENCED	MARY	NALEVO	D98	3-Jan-76	\$56,847	39485

```
DTR>
```

This shortcut also works if you type NOT before the EQUAL, EQ, =, or CONTAINING. It does not work, however, for any other specification including NOT_EQUAL.

You can form more complex RSEs by using the operators AND, OR, NOT, and BUT. The following example selects only employees who are trainees earning more than \$30,000 annually:

```
DTR> PRINT PERSONNEL WITH SALARY > 30000 AND STATUS = "TRAINEE"(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
12643	TRAINEE	JEFF	TASHKENT	C82	4-Apr-81	\$32,918	87465
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	87289

```
DTR>
```

6.3 Examples of Displaying Data with the PRINT Statement

This section shows you the variety of PRINT statements you can use to form record streams. Try creating RSEs of your own.



MK-01104-00

DTR> PRINT PERSONNEL WITH DEPT = "T32" AND SALARY BT 30000 AND 40000€

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUF ID
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	872E
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	872E

DTR> PRINT PERSONNEL WITH STATUS NE "EXPERIENCED" (RET)

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUF ID
12345	TRAINEE	HENRIETTA	MILLER				0000
12643	TRAINEE	JEFF	TASHKENT	C82	4-Apr-81	\$32,918	874E
32432	TRAINEE	THOMAS	SCHWEIK	F11	7-Nov-81	\$26,723	008E
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	872E
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	872E
49843	TRAINEE	BART	HAMMER	D98	4-Aug-81	\$26,392	394E
75639	TRAINEE	MIKE	MILLER	T44	6-Feb-81	\$16,750	0001
75640	TRAINEE	BEN	PUTNAM	T44	6-Feb-81	\$16,500	0001
87701	TRAINEE	NATHANIEL	CHONTZ	F11	28-Jan-82	\$24,502	008E
91023	TRAINEE	STAN	WITTGEN	G20	23-Dec-81	\$25,023	872E

DTR> PRINT PERSONNEL WITH DEPT CONT "T" BUT DEPT NE "TOP" (RET)

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUF ID
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	872E
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	0001
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	872E
75639	TRAINEE	MIKE	MILLER	T44	6-Feb-81	\$16,750	0001
75640	TRAINEE	BEN	PUTNAM	T44	6-Feb-81	\$16,500	0001
83764	EXPERIENCED	JIM	MEADER	T32	4-Apr-80	\$41,029	872E
87422	EXPERIENCED	MARTY	KELLEY	T32	26-Aug-82	\$18,750	872E

DTR>

You could have stated these examples in different terms. PERSONNEL WITH STATUS = "TRAINEE", for example, forms the same group of records as PERSONNEL WITH STATUS NE "EXPERIENCED". You can form a group of all the records in Department T32 with any of the following:

- DEPT = "T32"
- DEPT CONT "T" BUT DEPT NOT EQUAL "TOP", "T44"
- DEPT NOT EQUAL "TOP", "F11", "D98", "C82", "E46", "G20", "T44"
- ID = 34456,38462,48573,83764,87422
- LAST_NAME = "MORRISON", "SWAY", "KELLER", "MEADER", "KELLEY"

Obviously, the first example is the easiest and most straightforward. The important thing to remember is that many options are available to you when you form record streams. You can specify a record or group of records in many ways, as long as you use correct DATATRIEVE syntax.

6.4 Sorting Records in Record Streams

By adding a SORTED BY clause to the end of an RSE, you can tell DATATRIEVE to sort records in the record stream in the order you specify. A field that forms a basis for sorting is called a sort key.

For example, you can print the five records of the record stream PERSONNEL WITH DEPT = "T32" in alphabetical order by LAST_NAME:

```
DTR> PRINT PERSONNEL WITH DEPT = "T32" SORTED BY LAST_NAME(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	87289
87422	EXPERIENCED	MARTY	KELLEY	T32	26-Aug-82	\$18,750	87289
83764	EXPERIENCED	JIM	MEADER	T32	4-Apr-80	\$41,029	87289
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	87289
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	00012

```
DTR>
```

You can sort records by the values in several fields, separating each sort key from the next by a comma. DATATRIEVE sorts the records by the order in which you list the sort keys. Here is a record stream sorted first by DEPT, then by LAST_NAME:

```
DTR> PRINT PERSONNEL SORTED BY DEPT, LAST_NAME(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
12345	TRAINEE	HENRIETTA	MILLER				00000
90342	EXPERIENCED	BRUNO	DONCHIKOV	C82	9-Aug-78	\$35,952	87465
87465	EXPERIENCED	ANTHONY	IACOBONE	C82	2-Jan-73	\$58,462	00012
99029	EXPERIENCED	RANDY	PODERESIAN	C82	24-May-79	\$33,738	87465
49001	EXPERIENCED	DAN	ROBERTS	C82	7-Jul-79	\$41,395	87465
12643	TRAINEE	JEFF	TASHKENT	C82	4-Apr-81	\$32,918	87465
49843	TRAINEE	BART	HAMMER	D98	4-Aug-81	\$26,392	39485
84375	EXPERIENCED	MARY	NALEVO	D98	3-Jan-76	\$56,847	39485
39485	EXPERIENCED	DEE	TERRICK	D98	2-May-77	\$55,829	00012
02943	EXPERIENCED	CASS	TERRY	D98	2-Jan-80	\$29,908	39485
48475	EXPERIENCED	GAIL	CASSIDY	E46	2-May-78	\$55,407	00012
38465	EXPERIENCED	JOANNE	FREIBURG	E46	20-Feb-80	\$23,908	48475
87701	TRAINEE	NATHANIEL	CHONTZ	F11	28-Jan-82	\$24,502	00891
78923	EXPERIENCED	LYDIA	HARRISON	F11	19-Jun-79	\$40,747	00891
00891	EXPERIENCED	FRED	HOWL	F11	9-Apr-76	\$59,594	00012
32432	TRAINEE	THOMAS	SCHWEIK	F11	7-Nov-81	\$26,723	00891
87289	EXPERIENCED	LOUISE	DEPALMA	G20	28-Feb-79	\$57,598	00012
88001	EXPERIENCED	DAVID	LITELLA	G20	11-Nov-80	\$34,933	87289
91023	TRAINEE	STAN	WITTGEN	G20	23-Dec-81	\$25,023	87289
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	87289
87422	EXPERIENCED	MARTY	KELLEY	T32	26-Aug-82	\$18,750	87289
83764	EXPERIENCED	JIM	MEADER	T32	4-Apr-80	\$41,029	87289
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	87289
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	00012
75639	TRAINEE	MIKE	MILLER	T44	6-Feb-81	\$16,750	00012
75640	TRAINEE	BEN	PUTNAM	T44	6-Feb-81	\$16,500	00012
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	00012

```
DTR>
```

In this example DATATRIEVE sorts the records and then lists them in the following order:

- Henrietta Miller, the only employee with no department number
- All the employees in Department C82 before any employees in any other department
- Within Department C82, in alphabetical order by last name

DATATRIEVE then repeats this sequence for Department D98, E46, and so on.

Unless you specify otherwise, DATATRIEVE sorts records by the ascending value of the sort keys you include in the record selection expression. That is, it puts the record with the smallest value first and the largest value last. You can also print records in descending order by typing DESCENDING (or DESC) before a sort key:

```
DTR> PRINT PERSONNEL WITH DEPT = "T32" SORTED BY DESC SALARY(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	0001
83764	EXPERIENCED	JIM	MEADER	T32	4-Apr-80	\$41,029	8728
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	8728
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	8728
87422	EXPERIENCED	MARTY	KELLEY	T32	26-Aug-82	\$18,750	8728

```
DTR>
```

The following rules apply to the sort order of your lists:

- ASCENDING (or ASC) is the default sort order, sorting from smallest to largest. The default sort order is the order that DATATRIEVE uses to sort records if you do not specify a different order.
- Numbers have lower values than letters, so if you do not specify differently, DATATRIEVE first orders the data from 0 to 9, then A to Z.
- DESCENDING (or DESC) changes the sort order, sorting from largest to smallest.

You can change the sort order as often as you like in any list by specifying ASCENDING or DESCENDING before the name of the field whose order you want to change. When you change the sort order, DATATRIEVE will sort the following items in the new order until you specifically change the order back.

DTR> PRINT PERSONNEL SORTED BY DESC DEPT, ASCENDING SALARY(RET)

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	00012
75640	TRAINEE	BEN	PUTNAM	T44	6-Feb-81	\$16,500	00012
75639	TRAINEE	MIKE	MILLER	T44	6-Feb-81	\$16,750	00012
87422	EXPERIENCED	MARTY	KELLEY	T32	26-Aug-82	\$18,750	87289
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	87289
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	87289
83764	EXPERIENCED	JIM	MEADER	T32	4-Apr-80	\$41,029	87289
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	00012
91023	TRAINEE	STAN	WITTGEN	G20	23-Dec-81	\$25,023	87289
88001	EXPERIENCED	DAVID	LITELLA	G20	11-Nov-80	\$34,933	87289
87289	EXPERIENCED	LOUISE	DEPALMA	G20	28-Feb-79	\$57,598	00012
87701	TRAINEE	NATHANIEL	CHONTZ	F11	28-Jan-82	\$24,502	00891
32432	TRAINEE	THOMAS	SCHWEIK	F11	7-Nov-81	\$26,723	00891
78923	EXPERIENCED	LYDIA	HARRISON	F11	19-Jun-79	\$40,747	00891
00891	EXPERIENCED	FRED	HOWL	F11	9-Apr-76	\$59,594	00012
38465	EXPERIENCED	JOANNE	FREIBURG	E46	20-Feb-80	\$23,908	48475
48475	EXPERIENCED	GAIL	CASSIDY	E46	2-May-78	\$55,407	00012
49843	TRAINEE	BART	HAMMER	D98	4-Aug-81	\$26,392	39485
02943	EXPERIENCED	CASS	TERRY	D98	2-Jan-80	\$29,908	39485
39485	EXPERIENCED	DEE	TERRICK	D98	2-May-77	\$55,829	00012
84375	EXPERIENCED	MARY	NALEVO	D98	3-Jan-76	\$56,847	39485
12643	TRAINEE	JEFF	TASHKENT	C82	4-Apr-81	\$32,918	87465
99029	EXPERIENCED	RANDY	PODERESIAN	C82	24-May-79	\$33,738	87465
90342	EXPERIENCED	BRUNO	DONCHIKOV	C82	9-Aug-78	\$35,952	87465
49001	EXPERIENCED	DAN	ROBERTS	C82	7-Jul-79	\$41,395	87465
87465	EXPERIENCED	ANTHONY	IACOBONE	C82	2-Jan-73	\$58,462	00012
12345	TRAINEE	HENRIETTA	MILLER				00000

DTR>

6.5 Displaying Fields from Record Streams

You do not have to print all the fields in a record when you form a record stream with an RSE. To limit the number of fields to be printed, type PRINT followed by the names of the fields to be printed, the keyword OF, and the RSE.

Separate each field name from the next by a comma. This example shows that you can specify group fields and mix them with elementary fields. NAME is a group field containing the elementary fields FIRST_NAME and LAST_NAME. ID and SUP_ID are elementary fields.

DTR> PRINT ID, NAME, SUP_ID OF PERSONNEL WITH DEPT = "T32"(RET)

ID	FIRST NAME	LAST NAME	SUP ID
34456	HANK	MORRISON	87289
38462	BILL	SWAY	00012
48573	SY	KELLER	87289
83764	JIM	MEADER	87289
87422	MARTY	KELLEY	87289

DTR>

The next example limits the number of records displayed to five by using a restriction clause (FIRST 5). It also uses a sort clause (SORTED BY SALARY).

If you specify both a restriction clause and a sort clause in the same RSE, DATATRIEVE sorts the record stream before imposing the restriction. Here DATATRIEVE sorts the PERSONNEL database by salary. It bypasses the employees making less than \$25,000 and prints the first five employees making more than that figure. Notice that when you continue your print statement to a second line, DATATRIEVE prompts you with CON>, the continuation prompt.

```
DTR> PRINT NAME, SALARY OF FIRST 5 PERSONNEL WITH(RET)
[Looking for Boolean expression]
CON> SALARY GT 25000 SORTED BY SALARY(RET)
```

FIRST NAME	LAST NAME	SALARY
STAN	WITTGEN	\$25,023
BART	HAMMER	\$26,392
THOMAS	SCHWEIK	\$26,723
CASS	TERRY	\$29,908
HANK	MORRISON	\$30,000

```
DTR>
```

The phrase “[Looking for Boolean expression]” that appears on the line before the CON> prompt is referring to the kind of expression shown in Table 6-1. A Boolean expression is an expression that DATATRIEVE evaluates to either “True” or “False.”

6.6 Stopping the Display of Records

If you do not want to wait for DATATRIEVE to display all the records in a record stream, you can end the display by typing a CTRL/C:

```
DTR> PRINT PERSONNEL(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUI ID
12345	TRAINEE	HENRIETTA	MILLER				0000
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	0000
00891	EXPERIENCED	FRED	HOWL	F11	9-Apr-76	\$59,594	0000
02943	EXPERIENCED	CASS	TERRY	D98	2-Jan-80	\$29,908	3948
12643	TRAINEE	JEFF	TASHKENT	C82	4-Apr-81	\$32,918	8748
32432	TRAINEE	THOMAS	SCHWEIK	F11	7-Nov-81	\$26,723	0088
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	8728
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	0000
38465	EXPERIENCED	JOANNE	FREIBURG	E46	20-Feb-80	\$23,908	4847
39485	EXPERIENCED	DEE	TERRICK	D98	2-May-77	\$55,829	0000
48475	EXPERIENCED	GAIL	CASSIDY	E46	2-May-78	\$55,407	0000
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	8728

```
^C
Execution terminated by operator
```

```
DTR>
```

You can enter the CTRL/C at any point. Your terminal displays it as “^C”, as shown in the example above.

6.7 Using Collections

The record streams you have formed so far have all been temporary. You have to retype the entire PRINT statement to display the same set of records a second time or to sort the record stream in a different way.

The FIND statement forms a special type of record stream called a collection. Like a record stream, a collection is a group of records you select from the data available in a domain, such as a collection of all personnel with salaries greater than \$40,000. Unlike a record stream, a collection remains available until you release it.

Collections differ from record streams in three important ways:

- A collection is available to you until you:
 - Form another collection with the same name with another FIND command
 - Release the collection with a RELEASE command (See Section 6.9.)
 - Release the domain from which you formed the collection with a FINISH command (See Section 6.10.)
 - Exit from DATATRIEVE
- You need not retype the RSE if you want to change the sort order of the records or display a subset of the records or fields in the collection.
- You can name a collection to keep it from disappearing when you form other collections.

6.7.1 Forming Collections

To form a collection, type FIND and a record selection expression (RSE), and press RETURN:

```
DTR> FIND PERSONNEL WITH DEPT = "T32"(RET)
[5 records found]
DTR>
```

When you enter the FIND statement, DATATRIEVE looks for records that meet the conditions specified in the RSE. When that search is complete, DATATRIEVE shows how many records it has found but does not display them.

6.7.2 Displaying the CURRENT Collection

DATATRIEVE gives the name CURRENT to a collection when you form it. When you form a new collection with another FIND command, the new collection replaces the old one as the CURRENT collection.

If you want to see the records in the CURRENT collection, you must use a PRINT statement. Four different PRINT statements display the same records if you have not chosen a particular record in any collection:

- PRINT ALL CURRENT
- PRINT CURRENT
- PRINT ALL
- PRINT

To see how the first three PRINT statements have identical effects, you can try them all to display the five records in the CURRENT collection you just formed of PERSONNEL from Department T32. The example here shows the output you get:

DTR> PRINT ALL CURRENT^(RET)

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	872E
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	0001
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	872E
83764	EXPERIENCED	JIM	MEADER	T32	4-Apr-80	\$41,029	872E
87422	EXPERIENCED	MARTY	KELLEY	T32	26-Aug-82	\$18,750	872E

DTR>

When you enter PRINT ALL, DATATRIEVE assumes you are referring to the CURRENT collection. If you have no CURRENT collection and enter PRINT ALL, DATATRIEVE displays this message on your terminal:

A current collection has not been established.

When you type PRINT by itself, DATATRIEVE first looks to see if you have chosen one particular record with the SELECT statement. If you have, DATATRIEVE prints it. If you have not, DATATRIEVE first displays a message on your terminal and then displays all the records in the CURRENT collection:

DTR> PRINT^(RET)

No record selected; printing whole collection

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	872B
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	0001
48573	TRAINEE	SY	KELLER	T32	2-Aug-81	\$31,546	872B
83764	EXPERIENCED	JIM	MEADER	T32	4-Apr-80	\$41,029	872B
87422	EXPERIENCED	MARTY	KELLEY	T32	26-Aug-82	\$18,750	872B

DTR>

See Section 7.1 for information on the SELECT statement.

6.7.3 Displaying Sorted Records in the CURRENT Collection

The sort order specified in the PRINT statement affects only the record stream formed by the PRINT statement. It has no effect on the CURRENT collection, which always remains the same.

To see that the original order of the CURRENT collection does not change, use the following FIND statement to form a collection of employees reporting to the supervisor with ID Number 00012.

```
DTR> FIND PERSONNEL WITH SUP_ID = 00012(RET)
[9 records found]
DTR> PRINT CURRENT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	00012
00891	EXPERIENCED	FRED	HOWL	F11	9-Apr-76	\$59,594	00012
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	00012
39485	EXPERIENCED	DEE	TERRICK	D98	2-May-77	\$55,829	00012
48475	EXPERIENCED	GAIL	CASSIDY	E46	2-May-78	\$55,407	00012
75639	TRAINEE	MIKE	MILLER	T44	6-Feb-81	\$16,750	00012
75640	TRAINEE	BEN	PUTNAM	T44	6-Feb-81	\$16,500	00012
87289	EXPERIENCED	LOUISE	DEPALMA	G20	28-Feb-79	\$57,598	00012
87465	EXPERIENCED	ANTHONY	IACOBONE	C82	2-Jan-73	\$58,462	00012

Notice that the collection is made up of records sorted by ID. Now sort the collection and print it in alphabetical order.

```
DTR> PRINT CURRENT SORTED BY LAST_NAME(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
48475	EXPERIENCED	GAIL	CASSIDY	E46	2-May-78	\$55,407	00012
87289	EXPERIENCED	LOUISE	DEPALMA	G20	28-Feb-79	\$57,598	00012
00891	EXPERIENCED	FRED	HOWL	F11	9-Apr-76	\$59,594	00012
87465	EXPERIENCED	ANTHONY	IACOBONE	C82	2-Jan-73	\$58,462	00012
75639	TRAINEE	MIKE	MILLER	T44	6-Feb-81	\$16,750	00012
75640	TRAINEE	BEN	PUTNAM	T44	6-Feb-81	\$16,500	00012
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	00012
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	00012
39485	EXPERIENCED	DEE	TERRICK	D98	2-May-77	\$55,829	00012

Finally, to see that the collection itself has not been changed, have DATATRIEVE print the collection again without sorting it.

```
DTR> PRINT CURRENT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	00012
00891	EXPERIENCED	FRED	HOWL	F11	9-Apr-76	\$59,594	00012
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	00012
39485	EXPERIENCED	DEE	TERRICK	D98	2-May-77	\$55,829	00012
48475	EXPERIENCED	GAIL	CASSIDY	E46	2-May-78	\$55,407	00012

(continued on next page)

```

75639 TRAINEE      MIKE      MILLER    T44      6-Feb-81   $16,750 0001
75640 TRAINEE      BEN       PUTNAM    T44      6-Feb-81   $16,500 0001
87289 EXPERIENCED LOUISE    DEPALMA   G20      28-Feb-79  $57,598 0001
87465 EXPERIENCED ANTHONY   IACOBONE  CB2      2-Jan-73   $58,462 0001

```

DTR>

You can specify a sort order with the FIND statement just as you do with the PRINT statement. As for collections that do not specify a sort order, a PRINT statement specifying a sort order has no effect on the order in the collection itself.

6.7.4 Displaying Information About the CURRENT Collection

You can use the SHOW command to display information about the CURRENT collection. Type SHOW CURRENT and press RETURN:

```

DTR> SHOW CURRENT(RET)
Collection CURRENT
      Domain: PERSONNEL
      Number of Records: 9
      No Selected Record

```

DTR>

This display tells you that the CURRENT collection comes from the PERSONNEL domain, contains nine records, and has no selected record.

Chapter 7 explains selected records.

6.7.5 Making a New CURRENT Collection from an Old One

You can form a series of CURRENT collections, each based on the previous one. For example, you can keep changing the limiting value of the same field:

```

DTR> FIND PERSONNEL WITH SALARY > 30000(RET)
[16 records found]
DTR> FIND CURRENT WITH SALARY > 40000(RET)
[11 records found]
DTR> FIND CURRENT WITH SALARY > 50000(RET)
[8 records found]
DTR> FIND CURRENT WITH SALARY > 60000(RET)
[1 record found]
DTR> PRINT ALL(RET)

```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	0001

DTR>

You can also use a different field each time to reduce the number of records in the CURRENT collection:

```

DTR> FIND PERSONNEL WITH SUP_ID = 00012(RET)
[9 records found]

```

(continued on next page)

```
DTR> FIND CURRENT WITH SALARY > 57500(RET)
[4 records found]
DTR> FIND CURRENT WITH START_DATE GT "1-JAN-1978"(RET)
[1 record found]
DTR> PRINT CURRENT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
87289	EXPERIENCED	LOUISE	DEPALMA	G20	28-Feb-79	\$57,598	00012

```
DTR>
```

6.7.6 Using the CURRENT Collection in RSEs

When you want to refer to the CURRENT collection, you can type CURRENT instead of the original RSE you entered to form the collection. The following sequence forms a collection, prints it, sorts the records into a new order, and prints the first five in that order:

```
DTR> FIND PERSONNEL WITH SUP_ID = 00012(RET)
[9 records found]
DTR> PRINT ALL(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	00012
00891	EXPERIENCED	FRED	HOWL	F11	9-Apr-76	\$59,594	00012
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	00012
39485	EXPERIENCED	DEE	TERRICK	D98	2-May-77	\$55,829	00012
48475	EXPERIENCED	GAIL	CASSIDY	E46	2-May-78	\$55,407	00012
75639	TRAINEE	MIKE	MILLER	T44	6-Feb-81	\$16,750	00012
75640	TRAINEE	BEN	PUTNAM	T44	6-Feb-81	\$16,500	00012
87289	EXPERIENCED	LOUISE	DEPALMA	G20	28-Feb-79	\$57,598	00012
87465	EXPERIENCED	ANTHONY	IACOBONE	C82	2-Jan-73	\$58,462	00012

```
DTR> PRINT FIRST 5 CURRENT SORTED BY DESC SALARY(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	00012
00891	EXPERIENCED	FRED	HOWL	F11	9-Apr-76	\$59,594	00012
87465	EXPERIENCED	ANTHONY	IACOBONE	C82	2-Jan-73	\$58,462	00012
87289	EXPERIENCED	LOUISE	DEPALMA	G20	28-Feb-79	\$57,598	00012
39485	EXPERIENCED	DEE	TERRICK	D98	2-May-77	\$55,829	00012

```
DTR>
```

After this sequence of statements, the CURRENT collection remains the same. It contains all the original records in their original order.

6.8 Naming a Collection

So far, you have created only one collection at any one time — the CURRENT collection. Each time you formed a new CURRENT collection, the old one disappeared.

You can create more than one collection by giving each collection a name when you form it with the FIND statement. For example, you can form a collection of trainees and name the collection TYROS:

```
DTR> FIND TYROS IN PERSONNEL WITH STATUS = "TRAINEE"␣  
[10 records found]  
DTR>
```



MK-01102-00

This new CURRENT collection has the name TYROS. You can use any name you like. To confirm that the CURRENT collection has the name you gave it, use the SHOW COLLECTIONS command.

```
DTR> SHOW COLLECTIONS␣  
Collections:  
      TYROS (also CURRENT)  
DTR>
```

You can now use the collection name TYROS in other DATATRIEVE commands and statements. When you form a new CURRENT collection, the TYROS collection remains. It does not disappear as an unnamed collection would have. The name CURRENT becomes attached to the newly formed collection. You can still use TYROS to refer to the original collection in other DATATRIEVE commands and statements. The SHOW CURRENT command does not refer to TYROS any longer, but the SHOW TYROS command shows that it is still available:

```
DTR> FIND EXPERTS IN PERSONNEL WITH STATUS = "EXPERIENCED"␣  
[17 records found]  
  
DTR> SHOW CURRENT␣  
Collection CURRENT  
      Domain: PERSONNEL  
      Number of Records: 17  
      No Selected Record
```

(continued on next page)

```
DTR> SHOW TYROS(RET)
Collection TYROS
  Domain: PERSONNEL
  Number of Records: 10
  No Selected Record
```

```
DTR> PRINT FIRST 3 CURRENT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	00012
00891	EXPERIENCED	FRED	HOWL	F11	9-Apr-76	\$59,594	00012
02943	EXPERIENCED	CASS	TERRY	D98	2-Jan-80	\$29,908	39485

```
DTR> PRINT FIRST 3 TYROS(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
12345	TRAINEE	HENRIETTA	MILLER				00000
12643	TRAINEE	JEFF	TASHKENT	C82	4-Apr-81	\$32,918	87465
32432	TRAINEE	THOMAS	SCHWEIK	F11	7-Nov-81	\$26,723	00891

```
DTR>
```

6.8.1 Showing the Names of Existing Collections

You can have as many named collections as you want. To see the names of all the collections you have, use the SHOW COLLECTIONS command:

```
DTR> SHOW COLLECTIONS(RET)
Collections:
  EXPERTS (also CURRENT)
  TYROS
```

```
DTR>
```

In response to the SHOW COLLECTIONS command, DATATRIEVE lists the most recently formed collection first and the oldest of the existing collections last.

If the CURRENT collection has a name, DATATRIEVE displays a note beside the name of that collection to show that it is also the CURRENT collection. If the CURRENT collection is unnamed, DATATRIEVE displays the name CURRENT on the list of collections:

```
DTR> FIND PERSONNEL WITH DEPT = "C82"(RET)
[5 records found]
DTR> SHOW COLLECTIONS(RET)
Collections:
  CURRENT
  EXPERTS
  TYROS
```

(continued on next page)

DTR> PRINT FIRST 2 CURRENT^(RET)

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUF ID
12643	TRAINEE	JEFF	TASHKENT	C82	4-Apr-81	\$32,918	874E
49001	EXPERIENCED	DAN	ROBERTS	C82	7-Jul-79	\$41,395	874E

DTR>

To display records in a named collection that is not the CURRENT collection, you must use the collection name in the RSE of a PRINT statement:

DTR> PRINT FIRST 2 TYROS^(RET)

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUF ID
12345	TRAINEE	HENRIETTA	MILLER				000C
12643	TRAINEE	JEFF	TASHKENT	C82	4-Apr-81	\$32,918	874E

DTR> PRINT FIRST 2 EXPERTS^(RET)

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUF ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	0001
00891	EXPERIENCED	FRED	HOWL	F11	9-Apr-76	\$59,594	0001

DTR>

When you have access to more than one collection, PRINT ALL always causes DATATRIEVE to display the records in the CURRENT collection.

If you use PRINT ALL TYROS, or the name of any other collection you have formed, DATATRIEVE displays all the records in the collection you name.

6.8.2 Naming an Unnamed Collection

If you have not named your CURRENT collection and then decide you want to keep it, you can name it by using the FIND command:

DTR> FIND DEPT_C82 IN CURRENT^(RET)

[5 records found]

DTR> SHOW COLLECTIONS^(RET)

Collections:

DEPT_C82 (also CURRENT)
EXPERTS
TYROS

DTR>

This command does not actually rename the CURRENT collection. It forms a new CURRENT collection named DEPT_C82 that contains the same records as the previous CURRENT collection, which no longer exists.

You can also change the name of a named collection. The newly formed collection does not replace the older one. The new one goes, instead, to the top of the list of existing collections. The collection now has two names, C82 and CURRENT.

```
DTR> FIND C82 IN DEPT_C82(RET)
[5 records found]
DTR> SHOW COLLECTIONS(RET)
Collections:
    C82 (also CURRENT)
    DEPT_C82
    EXPERTS
    TYROS
```

DTR>

You can use this same method to impose new limits on a collection without having to retype the original RSE. The collection TYROS consists of all the employees who are trainees. You can form a new collection called NEW_TRAINEES by including only those TYROS that started after 1-Jan-1982. NEW_TRAINEES becomes the CURRENT collection, but you still have access to TYROS:

```
DTR> FIND NEW_TRAINEES IN TYROS WITH START_DATE GT "1-Jan-1982"(RET)
[2 records found]
DTR> SHOW COLLECTIONS(RET)
Collections:
    NEW_TRAINEES (also CURRENT)
    C82
    DEPT_C82
    EXPERTS
    TYROS
```

DTR>

6.8.3 Changing the Content of a Named Collection

To change the content of a named collection, use the same name in a FIND command with new conditions specified in the RSE. When you use the name of an existing collection, DATATRIEVE releases the old collection with the same name and makes the new one the CURRENT collection:

```
DTR> FIND NEW_TRAINEES IN TYROS WITH START_DATE GT "1-JUL-1981"(RET)
[6 records found]
DTR> SHOW COLLECTIONS(RET)
Collections:
    NEW_TRAINEES (also CURRENT)
    C82
    DEPT_C82
    EXPERTS
    TYROS
```

```
DTR> FIND NEW_TRAINEES IN TYROS WITH START_DATE GT "1-JAN-1982"(RET)
[2 records found]
DTR> SHOW COLLECTIONS(RET)
Collections:
    NEW_TRAINEES (also CURRENT)
    C82
    DEPT_C82
    EXPERTS
    TYROS
```

(continued on next page)

```
DTR> PRINT NEW_TRAINEES(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUF ID
34456	TRAINEE	HANK	MORRISON	T32	1-Mar-82	\$30,000	8728
87701	TRAINEE	NATHANIEL	CHONTZ	F11	28-Jan-82	\$24,502	0088

```
DTR>
```

6.9 Releasing Collections

When you no longer need to use a collection, you can release it with the **RELEASE** command. Releasing a collection does not affect the data file.

You can verify the release of a collection with the **SHOW COLLECTIONS** command:

```
DTR> RELEASE C82(RET)
```

```
DTR> SHOW COLLECTIONS(RET)
```

```
Collections:
    NEW_TRAINEES (also CURRENT)
    DEPT_C82
    EXPERTS
    TYROS
```

```
DTR> RELEASE CURRENT(RET)
```

```
DTR> SHOW COLLECTIONS(RET)
```

```
Collections:
    DEPT_C82 (also CURRENT)
    EXPERTS
    TYROS
```

```
DTR> RELEASE TYROS(RET)
```

```
DTR> SHOW COLLECTIONS(RET)
```

```
Collections:
    DEPT_C82 (also CURRENT)
    EXPERTS
```

```
DTR> RELEASE EXPERTS, DEPT_C82(RET)
```

```
DTR> SHOW COLLECTIONS(RET)
```

```
No established collections
```

```
DTR> RELEASE(RET)
```

```
Looking for Dictionary Name
```

```
CON> ^Z
```

```
DTR>
```

From this sequence, you can see several rules for releasing collections:

- You can release collections in any order.
- You can release more than one collection at a time.
- If you release the **CURRENT** collection, the next most recently formed collection becomes the new **CURRENT** collection.

- If you release a named current collection, you can refer to it in the **RELEASE** command by its name or by **CURRENT**.
- If you use the **RELEASE** command without specifying a collection or collections, **DATATRIEVE** prompts you for a collection with the **CON>** prompt.

6.10 Using the **FINISH** Command

You can use the **FINISH** command to release all your collections.

The **FINISH** command also ends your access to all your readied domains, or to specific domains.

To show the use of the **FINISH** command, the following example forms three collections from the **PERSONNEL** domain. The **SHOW COLLECTIONS** command lists the three collections, and the **SHOW READY** command lists **PERSONNEL** as a readied domain. After the **FINISH** command, the **SHOW COLLECTIONS** and **SHOW READY** commands show the effect of finishing the domain:

```
DTR> FIND TYROS IN PERSONNEL WITH STATUS = "TRAINEE" (RET)
[10 records found]
DTR> FIND EXPERTS IN PERSONNEL WITH STATUS = "EXPERIENCED" (RET)
[17 records found]
DTR> FIND C82 IN PERSONNEL WITH DEPT = "C82" (RET)
[5 records found]
DTR> SHOW COLLECTIONS (RET)
Collections:
      C82 (also CURRENT)
      EXPERTS
      TYROS

DTR> SHOW READY (RET)
Ready domains:
      PERSONNEL: RMS INDEXED, PROTECTED READ

DTR> FINISH (RET)
DTR> SHOW COLLECTIONS (RET)
No established collections

DTR> SHOW READY (RET)
No ready domains

DTR>
```

If you have more than one domain readied, you can finish specific domains by including the domain names in the **FINISH** command. To finish only the **PERSONNEL** domain, you type this command:

```
DTR> FINISH PERSONNEL (RET)
```

This command releases the **PERSONNEL** domain and any collections associated with it, but no others.

Modifying Data 7

When information changes or you have made errors in entering data, you need a way to enter the corrected information. In DATATRIEVE, you can make changes to existing records with the MODIFY statement.

You need to specify which records you want to change by isolating them either in a collection or in a record stream. This chapter discusses the use of both collections and record streams for modifying data.

7.1 Using the SELECT Statement with a Collection

To modify one or more records, you can form a collection and use the SELECT statement to choose a record you want to change. The following examples show how the SELECT statement chooses one record in a collection.

Ready the PERSONNEL domain and form a collection with the FIND statement. Display the collection on your terminal by using the PRINT ALL statement:

```
DTR> READY PERSONNEL MODIFY(RET)
DTR> FIND PERSONNEL WITH SUP_ID = 39485(RET)
[3 records found]
DTR> PRINT ALL(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
02943	EXPERIENCED	CASS	TERRY	D98	2-Jan-80	\$29,908	39485
49843	TRAINEE	BART	HAMMER	D98	4-Aug-81	\$26,392	39485
84375	EXPERIENCED	MARY	NALEVO	D98	3-Jan-76	\$56,847	39485

```
DTR>
```

When you use the SELECT statement, DATATRIEVE marks one record in the target collection. If you have not yet selected a record in a collection and do not specify a particular record with the SELECT statement, DATATRIEVE selects the first record as the target record. If you are at any other place in the collection, DATATRIEVE selects the next record.

To display the first record in the collection, enter:

```
DTR> SELECT(RET)
DTR> PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUI ID
02943	EXPERIENCED	CASS	TERRY	D98	2-Jan-80	\$29,908	3946

```
DTR>
```

When you enter another SELECT statement, DATATRIVE then marks the next record in the collection as the target record:

```
DTR> SELECT(RET)
DTR> PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUI ID
49843	TRAINEE	BART	HAMMER	D98	4-Aug-81	\$26,392	3946

```
DTR> SELECT(RET)
DTR> PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUI ID
84375	EXPERIENCED	MARY	NALEVO	D98	3-Jan-76	\$56,847	3946

```
DTR>
```

You can continue selecting records in an entire collection in this way, no matter how many records are present.

You can also display only part of the information in the selected record:

```
DTR> PRINT ID, NAME, START_DATE(RET)
```

ID	FIRST NAME	LAST NAME	START DATE
84375	MARY	NALEVO	3-Jan-76

```
DTR>
```

7.2 Using Other Forms of the SELECT Statement

When you have a collection, you can choose any single record in the collection with some form of the SELECT statement:

- **SELECT FIRST** – Selects the first record in the collection.
- **SELECT LAST** – Selects the last record in the collection.

- **SELECT** or **SELECT NEXT** – Selects the next record in the collection. If you have not selected any record, **SELECT NEXT** picks the first record. If there is no next record because you are at the end of the collection, **DATATRIEVE** displays the following error message:

```
Record number out of range for collection
Execution failed
```

- **SELECT n** – Selects the record in the collection indicated by the number *n*, which must be between 1 and the number of the last record in the collection. **SELECT 5**, for example, selects the fifth record in the collection. It does not select five records.

If you specify a number greater than the number of records in the collection, **DATATRIEVE** displays the following error message:

```
Record number out of range for collection
Execution failed
```

The following series of **SELECT** and **PRINT** statements shows how the various forms of the **SELECT** statement work. The examples also show how to include two or more **DATATRIEVE** statements on a single input line by separating the statements with a semicolon (;).

```
DTR> SELECT FIRST; PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
02943	EXPERIENCED	CASS	TERRY	D98	2-Jan-80	\$29,908	39485

```
DTR> SELECT LAST; PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
84375	EXPERIENCED	MARY	NALEVO	D98	3-Jan-76	\$56,847	39485

```
DTR> SELECT 1; PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
02943	EXPERIENCED	CASS	TERRY	D98	2-Jan-80	\$29,908	39485

```
DTR> SELECT NEXT; PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
49843	TRAINEE	BART	HAMMER	D98	4-Aug-81	\$26,392	39485

```
DTR> SELECT; PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
84375	EXPERIENCED	MARY	NALEVO	D98	3-Jan-76	\$56,847	39485

```
DTR>
```

7.3 Using the SHOW Command to Identify the Selected Record

You can use the **SHOW** command to find out which record in a collection is the selected record. Type **SHOW**, followed by the collection name. **DATATRIEVE** returns this information, showing that the third record is the selected record in the **CURRENT** collection:

```
DTR> SHOW CURRENT(RET)
Collection CURRENT
  Domain: PERSONNEL
  Number of Records: 3
  Selected Record: 3
```

```
DTR>
```



MK-01106-00

7.4 Modifying a Selected Record in a Collection

To update or correct information in a selected record of a collection, follow these steps:

1. **READY** the domain for **MODIFY** access.
2. Form a collection of records with the **FIND** statement.
3. Use the **SELECT** statement to isolate the record you want to change.

4. Use the PRINT statement to check that the selected record is the one you intend to change.
5. Use the MODIFY statement to change the selected record.
6. Use the PRINT statement to check that the record has been changed correctly.

When you have selected a record that you want to modify in a collection, enter the keyword MODIFY. DATATRIEVE prompts you for a value for each elementary field in the record. You cannot modify the value of the primary key field, ID. Use the TAB to retain the values of any fields you want to leave unchanged. Use the SPACE to enter a blank or a zero depending on the field definition.

In the following example, an employee is transferring to a different department under a new supervisor and at an increased salary:

```
DTR> READY PERSONNEL MODIFY(RET)
DTR> FIND PERSONNEL WITH LAST_NAME = "HARRISON"(RET)
[1 record found]
DTR> SELECT; PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
78923	EXPERIENCED	LYDIA	HARRISON	F11	19-Jun-79	\$40,747	00891

```
DTR> MODIFY(RET)
Enter ID: (TAB)(RET)
Enter EMPLOYEE_STATUS: (TAB)(RET)
Enter FIRST_NAME: (TAB)(RET)
Enter LAST_NAME: (TAB)(RET)
Enter DEPT: E46(RET)
Enter START_DATE: (TAB)(RET)
Enter SALARY: 44590(RET)
Enter SUP_ID: 00012(RET)
DTR> PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
78923	EXPERIENCED	LYDIA	HARRISON	E46	19-Jun-79	\$44,590	00012

DTR>

When you use the MODIFY statement, DATATRIEVE prompts you for your entry to each field of the record you want to modify. No modification of any part of the record takes place if you enter CTRL/Z before pressing RETURN at the end of the last field value prompt.

```
DTR> FIND PERSONNEL WITH ID = 38462(RET)
[1 record found]
DTR> SELECT; PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	00012

(continued on next page)

```

DTR> MODIFY(RET)
Enter ID: (TAB)(RET)
Enter EMPLOYEE_STATUS: (TAB)(RET)
Enter FIRST_NAME: WILLIAM(RET)
Enter LAST_NAME: (TAB)(RET)
Enter DEPT: E46(RET)
Enter START_DATE: (TAB)(RET)
Enter Salary: 58(RET)
Enter SUP_ID: ^Z
Execution terminated by operator
DTR>

```

Because you entered CTRL/Z, the permanent record has remained unchanged:

```

DTR> PRINT CURRENT(RET)

```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SU ID
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	000

```

DTR>

```

When you press RETURN instead of a value, TAB, or space in response to a MODIFY prompt, DATATRIEVE repeats the prompt until you enter something acceptable. The following example shows DATATRIEVE's responses to two other kinds of error:

```

DTR> FIND PERSONNEL WITH ID = 78923(RET)
[1 record found]
DTR> SELECT; PRINT(RET)

```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SL ID
78923	EXPERIENCED	LYDIA	HARRISON	E46	19-Jun-79	\$44,590	000

```

DTR> MODIFY(RET)
Enter ID: 123456(RET)
Truncation during assignment
Re-enter ID: 12345(RET)
Enter EMPLOYEE_STATUS: NOVICE(RET)
Validation error for EMPLOYEE_STATUS
Re-enter EMPLOYEE_STATUS: ^Z
Execution terminated by operator
DTR>

```

The record definition for PERSONNEL restricts the entries you can make in the EMPLOYEE_STATUS field to either EXPERIENCED or TRAINEE. It therefore rejects the entry "NOVICE." DATATRIEVE also rejects the 6-digit SUP_ID number, because the record definition restricts the field to five digits.

7.5 Modifying Fields in Indexed Data Files

You can modify any field of a record in a sequential file. If you are working with an indexed file, however, you cannot change the primary key field. You may not be allowed to modify other key fields, as well, depending on the file's characteristics.

Some changes are simpler to do by specifying only the fields you want to change. At times it is easier to tab past the fields you do not want to change, rather than specifying the fields you do want to change.

With the SHOW READY command, you can find out what type of data file is associated with a domain:

```
DTR> SHOW READY(RET)
Ready domains:
      PERSONNEL:  RMS INDEXED, PROTECTED MODIFY
DTR>
```

The word "indexed" tells you that the data associated with the PERSONNEL domain is in an indexed file. The SHOW FIELDS command tells you which fields are keys:

```
DTR> SHOW FIELDS(RET)
PERSONNEL
  PERSON
    ID      [Number, indexed key]
    EMPLOYEE_STATUS (STATUS)      [Character string]
    EMPLOYEE_NAME (NAME)
      FIRST_NAME (F_NAME) [Character string]
      LAST_NAME (L_NAME)  [Character string]
    DEPT    [Character string]
    START_DATE [Date]
    SALARY  [Number]
    SUP_ID  [Number]
DTR>
```

In the PERSONNEL record definition, the ID field is the primary key. If you try to modify ID, DATATRIEVE sends an error message:

```
DTR> READY PERSONNEL MODIFY(RET)
DTR> FIND PERSONNEL WITH LAST_NAME = "HARRISON"(RET)
[1 record found]
DTR> SELECT;PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
78923	EXPERIENCED	LYDIA	HARRISON	E46	19-Jun-79	\$44,590	00012

```
DTR> MODIFY ID(RET)
Enter ID: 74252(RET)
Attempt to update key field without change attribute
Execution failed
DTR>
```

If you try to modify a key field, DATATRIEVE does not print an error message until the entire modification statement is complete. Then, it cancels the entire modification statement, making none of the changes you requested, as the following example shows:

```
DTR> FIND PERSONNEL WITH ID = 78923(RET)
[1 record found]
```

(continued on next page)

```
DTR> SELECT;PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUF ID
78923	EXPERIENCED	LYDIA	HARRISON	E46	19-Jun-79	\$44,590	0001

```
DTR> MODIFY ID,DEPT,SALARY,SUP_ID(RET)
```

```
Enter ID: 78901(RET)
```

```
Enter DEPT: F11(RET)
```

```
Enter SALARY: 40747(RET)
```

```
Enter SUP_ID: 00891(RET)
```

```
Attempt to update Key field without change attribute
```

```
Execution failed
```

```
DTR> PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUF ID
78923	EXPERIENCED	LYDIA	HARRISON	E46	19-Jun-1979	\$44,590	0001

```
DTR>
```

If you specify a group field name in a MODIFY statement, DATATRIEVE prompts you for a value for each elementary field in the group. In this example, the query name for the group field EMPLOYEE_NAME is NAME, and includes both FIRST_NAME and LAST_NAME:

```
DTR> FIND PERSONNEL WITH LAST_NAME = "FREIBURG"(RET)
```

```
[1 record found]
```

```
DTR> SELECT; PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUF ID
38465	EXPERIENCED	JOANNE	FREIBURG	E46	20-Feb-80	\$23,908	4847

```
DTR> MODIFY NAME(RET)
```

```
Enter FIRST_NAME: JOANNE(RET)
```

```
Enter LAST_NAME: KNECHTBURG(RET)
```

```
DTR> PRINT ID, NAME(RET)
```

ID	FIRST NAME	LAST NAME
38465	JOANNE	KNECHTBURG

```
DTR>
```

7.6 Modifying All Records in a Collection

You can use MODIFY ALL to make changes in all the records of a collection. Before modifying an entire collection, be very certain that you do not want certain records exempted from the change. If you make a mistake, it can lead to a lot of work to replace the data you did not want to change. Do not be so afraid of MODIFY ALL that you do not use it when appropriate, but do be careful.

To change all the records in the CURRENT collection, you must first establish the target collection as the CURRENT collection.

```
DTR> READY PERSONNEL MODIFY(RET)
DTR> FIND PERSONNEL WITH SUP_ID = 39485(RET)
[3 records found]
DTR> PRINT ALL(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
02943	EXPERIENCED	CASS	TERRY	D98	2-Jan-80	\$29,908	39485
49843	TRAINEE	BART	HAMMER	D98	4-Aug-81	\$26,392	39485
84375	EXPERIENCED	MARY	NALEVO	D98	3-Jan-76	\$56,847	39485

```
DTR>
```

In the **MODIFY** statement, include the keyword **ALL** to specify that the change applies to all the records in the **CURRENT** collection, then list the fields you want to change.

```
DTR> MODIFY ALL DEPT, SUP_ID(RET)
Enter DEPT: C82(RET)
Enter SUP_ID: 87465(RET)
DTR> PRINT ALL(RET)
```

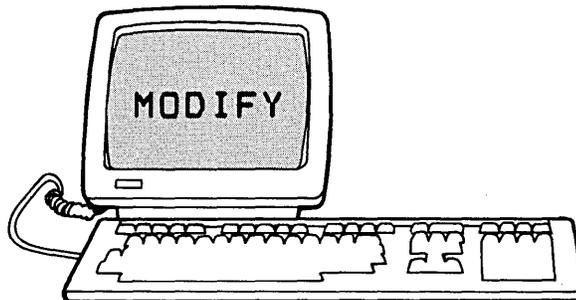
ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
02943	EXPERIENCED	CASS	TERRY	C82	2-Jan-80	\$29,908	87465
49843	TRAINEE	BART	HAMMER	C82	4-Aug-81	\$26,392	87465
84375	EXPERIENCED	MARY	NALEVO	C82	3-Jan-76	\$56,847	87465

```
DTR>
```

7.7 Modifying Records in a Record Stream

You can modify records without forming a collection or using a **SELECT** statement. You specify the records to be changed by putting the keywords **ALL OF** and a record selection expression (**RSE**) at the end of the **MODIFY** statement.

For **MODIFY** statements that change entire record streams, **DATATRIEVE** prompts only once for each elementary field indicated by the keyword **ALL** or by the list of fields. **DATATRIEVE** then changes all the records in the record stream using that single set of values. Consequently, you should avoid using a statement such as **MODIFY ALL OF PERSONNEL**.



MK-01101-00

7.8 Modifying Specific Fields in a Record Stream

To modify one field of the records in a record stream, specify the field name and the record selection expression in the MODIFY statement:

```
DTR> PRINT PERSONNEL WITH SUP_ID = 00891(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
32432	TRAINEE	THOMAS	SCHWEIK	F11	7-Nov-81	\$26,723	00891
87701	TRAINEE	NATHANIEL	CHONTZ	F11	28-Jan-82	\$24,502	00891

```
DTR> MODIFY SALARY OF FIRST 1 PERSONNEL WITH SUP_ID = 00891(RET)
```

```
Enter SALARY: 28605(RET)
```

```
DTR> PRINT PERSONNEL WITH SUP_ID = 00891(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
32432	TRAINEE	THOMAS	SCHWEIK	F11	7-Nov-81	\$28,605	00891
87701	TRAINEE	NATHANIEL	CHONTZ	F11	28-Jan-82	\$24,502	00891

```
DTR>
```

To modify two or more fields, list the fields before the OF rse portion of the statement, separating the fields with commas:

```
DTR> PRINT PERSONNEL WITH SUP_ID = 00891 AND SALARY > 25000(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
32432	TRAINEE	THOMAS	SCHWEIK	F11	7-Nov-81	\$28,605	00891

```
DTR> MODIFY DEPT, SUP_ID OF PERSONNEL WITH SUP_ID = 00891 AND SALARY > 25000(RET)
```

```
Enter DEPT: E46(RET)
```

```
Enter SUP_ID: 48475(RET)
```

```
DTR> PRINT PERSONNEL WITH DEPT = "E46"(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
32432	TRAINEE	THOMAS	SCHWEIK	E46	7-Nov-81	\$28,605	48475
38465	EXPERIENCED	JOANNE	KNECHTBURG	E46	20-Feb-80	\$23,908	48475
48475	EXPERIENCED	GAIL	CASSIDY	E46	2-May-78	\$55,407	00012
78923	EXPERIENCED	LYDIA	HARRISON	E46	19-Jun-79	\$44,590	00012

```
DTR>
```

Erasing Data 8

Sometimes you have to remove records from your data file. For an indexed data file like the one associated with the PERSONNEL domain, you can use the ERASE command. You cannot erase records from a sequential file. To remove data from a sequential file, however, you can use the MODIFY statement.

8.1 Using the ERASE Statement on Indexed Files

Because ERASE destroys records, use your judgment to prevent accidental loss of data.

To use the ERASE command, you must ready a domain for WRITE access:

```
DTR> READY PERSONNEL WRITE(RET)
DTR>
```

8.1.1 Erasing a Selected Record

To erase a selected record, follow these steps:

1. Establish a collection containing the record you want to erase.
2. Select that record.
3. Type ERASE and press RETURN.

The following example first creates and then erases a record in PERSONNEL.

```
DTR> READY PERSONNEL WRITE(RET)
DTR> STORE PERSONNEL(RET)
Enter ID: 00013(RET)
Enter EMPLOYEE_STATUS: TRAINEE(RET)
Enter FIRST_NAME: PETER(RET)
Enter LAST_NAME: SWAYZE(RET)
Enter DEPT: B17(RET)
Enter START_DATE: 1-Sep-1982(RET)
Enter SALARY: 20159(RET)
Enter SUP_ID: 87289(RET)
```

(continued on next page)

```
DTR> FIND PERSONNEL WITH DEPT = "B17"(RET)
[1 record found]
DTR> SELECT;PRINT(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00013	TRAINEE	PETER	SWAYZE	B17	1-Sep-82	\$20,159	8782

```
DTR> ERASE(RET)
DTR> PRINT(RET)
DTR> No record selected, printing whole collection
DTR>
```

DATATRIEVE prints no records because you have erased the only record in the collection. If the CURRENT collection had contained more than one record, the PRINT statement would have displayed the remaining records in the collection.

Even though you erase the only record in a collection, however, the collection does not disappear. The record has been erased from the data file and removed from the collection. Only its slot in the collection remains, and that slot is empty. The empty slot disappears only when the collection is released by the RELEASE, FINISH, or EXIT commands or when the collection is sorted or renamed.

```
DTR> SHOW CURRENT(RET)
Collection CURRENT
  Domain: PERSONNEL
  Number of Records: 1
  Selected Record: 1
```

```
DTR> FIND PERSONNEL WITH LAST_NAME = "SWAYZE"(RET)
[0 records found]
DTR>
```

8.1.2 Erasing a Collection of Records

To ERASE all the records in a collection, establish the collection and type ERASE ALL. Pay careful attention to the extent and content of the collection. Once you ERASE the records, you cannot recover them.

```
DTR> FIND PERSONNEL WITH DEPT = "G20"(RET)
[3 records found]
DTR> PRINT ALL(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
87289	EXPERIENCED	LOUISE	DEPALMA	G20	28-Feb-79	\$57,598	000
88001	EXPERIENCED	DAVID	LITELLA	G20	11-Nov-80	\$34,933	872
91023	TRAINEE	STAN	WITTGEN	G20	23-Dec-81	\$25,023	872

```
DTR> ERASE ALL(RET)
DTR> PRINT (RET)
No record selected, printing whole collection
DTR>
```

The PRINT statement displays no records because all the records in the collection have been erased and removed from the data file.

8.1.3 Erasing a Record Stream

To ERASE all the records in a record stream, specify the RSE in the ERASE ALL statement. As with collections, pay careful attention to the extent and content of the record stream.

```
DTR> PRINT PERSONNEL WITH DEPT = "F11"(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00891	EXPERIENCED	FRED	HOWL	F11	9-APR-76	\$59,594	00012
87701	TRAINEE	NATHANIEL	CHONTZ	F11	28-Jan-82	\$24,502	00891

```
DTR> ERASE ALL OF PERSONNEL WITH DEPT = "F11"(RET)
```

```
DTR> PRINT PERSONNEL WITH DEPT = "F11"(RET)
```

```
DTR>
```

8.2 Using the MODIFY Statement on Sequential Files

You cannot erase records from a sequential data file. You can, however, remove the data stored in records you want to eliminate from the file. Use the MODIFY statement to substitute a blank record for a record you want to remove from the data file.

Use any form of the MODIFY statement that chooses the records you want to eliminate. Respond to the prompts for field values by typing a space and pressing RETURN. The space changes the value of numeric fields to zero and the value of character fields to blank.

Writing Simple Reports 9

The DATATRIEVE Report Writer arranges your data in formats that are easy to read. You can control the format of the report or allow the Report Writer to do some or all the formatting automatically. This chapter introduces the basic formatting capabilities of the Report Writer.

9.1 The General Features of a Report

When you display records with the PRINT statement, you are creating a simple form of report:

```
DTR> READY PERSONNEL(RET)
DTR> FIND PERSONNEL WITH SUP_ID = 00012(RET)
[8 records found]
DTR> PRINT ALL(RET)
```

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	00012
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	00012
39485	EXPERIENCED	DEE	TERRICK	D98	2-May-77	\$55,829	00012
48475	EXPERIENCED	GAIL	CASSIDY	E46	2-May-78	\$55,407	00012
75639	TRAINEE	MIKE	MILLER	T44	6-Feb-81	\$16,750	00012
75640	TRAINEE	BEN	PUTNAM	T44	6-Feb-81	\$16,500	00012
78923	EXPERIENCED	LYDIA	HARRISON	E46	19-Jun-79	\$44,590	00012
87465	EXPERIENCED	ANTHONY	IACOBONE	C82	2-Jan-73	\$58,462	00012

```
DTR>
```

The PRINT statement gives you some control over the display of your data, but the Report Writer can do more formatting for you than the PRINT statement can.

Type the following commands and statements to produce a report based on the collection you have just printed using the PRINT statement. This report is the

simplest type you can create with the Report Writer. The RW> prompt shows when you are using the Report Writer:

```
DTR> FIND PERSONNEL WITH SUP_ID = 00012(RET)
[8 records found]
DTR> REPORT CURRENT(RET)
RW> PRINT PERSON(RET)
RW> END_REPORT(RET)
```

2-Nov-82
Page 1

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	00012
38462	EXPERIENCED	BILL	SWAY	T32	5-May-80	\$54,000	00012
39485	EXPERIENCED	DEE	TERRICK	D98	2-May-77	\$55,829	00012
48475	EXPERIENCED	GAIL	CASSIDY	E46	2-May-78	\$55,407	00012
75639	TRAINEE	MIKE	MILLER	T44	6-Feb-81	\$16,750	00012
75640	TRAINEE	BEN	PUTNAM	T44	6-Feb-81	\$16,500	00012
78923	EXPERIENCED	LYDIA	HARRISON	E46	19-Jun-79	\$44,590	00012
87465	EXPERIENCED	ANTHONY	IACOBONE	C82	2-Jan-73	\$58,462	00012

DTR>

“PERSON” is a group field, called a top-level field, that contains all the other fields of the PERSONNEL record. When you use the Report Writer PRINT statement, you cannot simply type PRINT by itself. You must type a field or fields after PRINT. When you enter the Report Writer statement “PRINT PERSON”, the Report Writer responds with all the information in the collection because the field PERSON contains all the fields in the record. If you wished, you could type PRINT SALARY or PRINT ID, SALARY.

Notice that the report does not display immediately after you type the PRINT statement. You must type END_REPORT in order to see the report.

This report, unlike the output of the PRINT statement, has the current date, a page number, and data columns evenly spaced across the entire page. If there are too many lines for a single page, the Report Writer skips automatically to the next page, printing the appropriate page number each time it starts a new page.

9.2 The Report Specification

You can create a collection before you type a REPORT statement, or you can form a record stream with an RSE in the REPORT statement itself. The collection or the record stream supplies the data for your report.

The series of statements that creates the report forms a unit called the report specification:

- The REPORT statement starts the report specification. As you will see later, you can specify an RSE in the REPORT statement.

- The Report Writer PRINT statement specifies the fields containing the data you want to report. In the example, PERSON is the group field that contains all the fields in the PERSONNEL record definition.
- To give a report a title, use the SET REPORT_NAME statement. Enclose the name within quotation marks. If you do not specify a name, DATATRIEVE prints the report without one.
- The END_REPORT command ends the report specification.

The following example shows you how to use the SET REPORT_NAME statement in the report specification:

```
DTR> FIND FIRST 1 PERSONNEL(RET)
[1 Record found]
DTR> REPORT CURRENT(RET)
RW> SET REPORT_NAME = "FIRST PERSONNEL RECORD"(RET)
RW> PRINT PERSON(RET)
RW> END_REPORT(RET)
```

FIRST PERSONNEL RECORD

16-Nov-82
Page 1

ID	STATUS	FIRST NAME	LAST NAME	DEPT	START DATE	SALARY	SUP ID
00012	EXPERIENCED	CHARLOTTE	SPIVA	TOP	12-Sep-72	\$75,892	00012

DTR>

If you do not type REPORT CURRENT but just REPORT, DATATRIEVE reports on the current collection.

9.2.1 Multiple-Line Report Names

To give a report a multiple-line title, enclose each line in quotation marks, and separate each line from the next with a slash. Report names must be less than 100 characters. The sample report specification in the following section uses the following multiple-line report name:

```
RW> SET REPORT_NAME = "TRAINEE EMPLOYEES"/"BY"/"DEPARTMENT AND SALARY"(RET)
```

9.2.2 The Columns_Page Option

By using the Report Writer's SET COLUMNS_PAGE statement, you can control the width of the report. The word "column" in the SET COLUMNS_PAGE statement refers to the number of spaces in the line, not to the columns of data in the report. The number of columns you specify must be between 1 and 255. If you do not specify a number, the Report Writer uses a default value of 80.

The next sample report specification uses this statement:

```
RW> SET COLUMNS_PAGE = 50(RET)
```

The REPORT command in this example specifies a record stream with the following RSE: CURRENT SORTED BY DEPT, SALARY. The PRINT statement specifies the fields ID, NAME, DEPT, SALARY:

```
DTR> FIND PERSONNEL WITH STATUS = "TRAINEE"(RET)
[8 records found]
DTR> REPORT CURRENT SORTED BY DEPT, SALARY(RET)
RW> SET REPORT_NAME = "TRAINEE EMPLOYEES"/"BY"/" DEPARTMENT AND SALARY"(RET)
RW> SET COLUMNS_PAGE = 50(RET)
RW> PRINT ID, NAME, DEPT, SALARY(RET)
RW> END_REPORT(RET)
```

TRAINEE EMPLOYEES
BY
DEPARTMENT AND SALARY 2-Nov-83
Page 1

ID	FIRST NAME	LAST NAME	DEPT	SALARY
12345	HENRIETTA	MILLER		
49843	BART	HAMMER	C82	\$26,392
12643	JEFF	TASHKENT	C82	\$32,918
32432	THOMAS	SCHWEIK	E46	\$28,605
34456	HANK	MORRISON	T32	\$30,000
48573	SY	KELLER	T32	\$31,546
75640	BEN	PUTNAM	T44	\$16,500
75639	MIKE	MILLER	T44	\$16,750

DTR>



MK-01107-00

If you make the COLUMNS_PAGE setting too small, the Report Writer moves the fields that do not fit to the next line of the display.

9.2.3 Print List Options

The Report Writer allows you to control many aspects of a report format. With the Report Writer you can:

- Create new fields, by computing new information from the data available
- Change column headers (the labels at the tops of columns)
- Create new column headers for new fields
- Print summary lines at the bottom of a page or the end of a report
- Print a count of the records making up the report
- Compute averages of numerical fields in the report
- Specify the format of lines in the report
- Specify some simple statistical functions

These are some of the options the Report Writer offers you. See the *DATATRIEVE-11 Guide to Writing Reports* for details of these and other options.

9.3 Output of Reports

Unless you work on a hard copy terminal, the reports created in the previous examples appear on your video screen and disappear as you continue working. To save a report, you can select an output device or file for it as a part of the REPORT statement.

9.3.1 Output to a Line Printer

To get a copy of your report from a line printer, type ON LP: at the end of the REPORT statement:

```
RW> REPORT ON LP:(RET)
```

If the REPORT statement contains an RSE, put the output specification after the RSE:

```
RW> REPORT PERSONNEL WITH DEPT = "G20" ON LP:(RET)
```

9.3.2 Output to a Disk File

If you want to store your report in a disk file, type the file specification after ON at the end of the REPORT statement. The following command creates a file called REPORT.TXT in your directory.

```
RW> REPORT TRAINEES ON REPORT.TXT(RET)
```

If you send your report to a disk file, you can use operating system utilities to print a hard copy of it at your convenience. You can also make it available for display on other terminals, or send it across a network link.

9.4 Saving Report Specifications

When you create a report you will want to use again, you can save the report specification by defining the report as part of a procedure. A procedure is a sequence of commands and statements that you can construct to perform a function you use frequently and want to avoid retyping each time you use it. Chapter 9 of the *DATATRIEVE-11 User's Guide* explains defining procedures. Chapter 1 of the *DATATRIEVE-11 Guide to Writing Reports* discusses using procedures to produce reports.

9.5 Summary

Report specifications can be complex because the Report Writer gives you so many ways to control the format. Mastery of report writing comes with patience and practice. Remember these seven basic characteristics of report writing with the DATATRIEVE Report Writer:

- You must READY the domain containing the data you want to report.
- The data you want to report must be contained in a collection or in the record stream you create in the REPORT statement.
- The REPORT statement gives you access to the Report Writer.
- The RW> prompt tells you to enter Report Writer statements.
- You can give your report a name with the SET REPORT-NAME statement. If you specify a name, you must enclose it in quotation marks.
- In the Report Writer PRINT statement, you must specify the fields containing information you want to report. You can have only one PRINT statement in a report specification.
- You must end the report specification with the END_REPORT statement.

With the statements presented in this chapter, you can now experiment with the PERSONNEL data or with data from a domain you have defined for yourself, and practice writing your own sample reports.

Glossary

Boolean operators

Symbols or phrases which enable you to join two or more expressions containing relational operators. Boolean operators are AND, OR, NOT, and BUT. For example, the expression SALARY < 60000 BUT SALARY NE 0 contains the Boolean operator "BUT". The symbols "<" and NE are relational operators.

Collection

A special type of record stream formed with the FIND statement. You can also name a collection. You can refer to a collection in subsequent statements until you replace it with another collection, release it with the RELEASE command, finish the domain from which it derives, or exit DATATRIEVE. You can have several named collections available at once.

Column headers

In a report or in the output of a PRINT statement, the headings that label the columns of data.

Command file

A text file containing DATATRIEVE commands and statements. You can create command files with the Application Design Tool (ADT), the EXTRACT command, or a text editor.

Data dictionary

An indexed file used to store DATATRIEVE definitions and access privileged information.

Data file

A collection of related data records.

Data record

A body of related information that is the basic unit for storing data. For example, the several items of data about an individual employee in the PERSONNEL domain make up a record for that employee.

Database

A collection of records maintained on a computer storage device and organized to facilitate immediate and accurate inquiry and update.

Detail lines

The lines containing data items in a report.

Domain

A data structure that associates a name with the relationship between a data file and a record definition. Using the domain name gives access to information in the data file as interpreted by the record definition.

Elementary field

A record segment containing one item of information.

Field

A segment of a data record. For example, each PERSONNEL record contains these fields: ID, employee status, employee name, department, starting date, salary, and the ID of the employee's supervisor.

Group field

A record segment containing one or more elementary fields. In PERSONNEL, EMPLOYEE_NAME is a group field containing the employee's first and last names.

Index key

A field that is the basis for the indexing of an indexed file. For the domain PERSONNEL, the index key is ID. DATATRIEVE indexes the records by identification numbers. You can use primary keys, which you cannot change, and alternate keys.

Indexed file

A data file that includes records and an index specifying the location of each record in the file.

Primary key

The index key whose value determines the order of records in an indexed file. You cannot modify or erase the value in a primary key field.

Procedure

A fixed sequence of commands, statements, clauses, or arguments that you create, name, and store in the data dictionary.

Query header

A substitute column header that replaces the field name when DATATRIEVE displays values from a field on your terminal. You cannot use query headers as query names.

Query name

A synonym for a field name, usually a shorter word to make input easier to type and remember. For example, you can type STATUS for EMPLOYEE_STATUS to refer to that field in DATATRIEVE's sample domain PERSONNEL.

Record definition

An entry in a data dictionary that describes the fields in a record.

Record selection expression (RSE)

A phrase defining specific conditions individual records must meet before they are included in a record stream.

Record stream

A temporary group of related records formed by a record selection expression.

Relational operators

Symbols or phrases you can use to compare values. For example, in the DATATRIEVE statement FIND PERSONNEL WITH SALARY > 50000, SALARY > 50000 is a Boolean expression containing the relational operator ">" (greater than).

Report header

The heading of a DATATRIEVE Report, consisting of an optional, centered report-name and, at the top-right corner of the report, a date and a page number.

Report specification

A series of Report Writer statements that creates a report and specifies its format.

Restriction clause

A phrase allowing you to specify the maximum number of records in a record stream.

Selected record

In a collection, the one record marked by the `SELECT` statement, and available for display or modifying without specifying a record selection expression.

Sequential file

A file whose records are physically stored next to each other in a computer's storage device. You cannot delete records from a sequential file with `DATATRIEVE`.

Sort key

A field that forms the basis for sorting. For example, you can rearrange the records in `DATATRIEVE`'s sample domain `PERSONNEL` according to department by using the sort key `DEPT`.

Index

<
 See LESS_THAN relational operator
>
 See GREATER_THAN relational operator

A

Accessing domains, 2-4 to 2-5
ADT command, 3-6
 See Also Application Design Tool
AND Boolean operator, 6-5
Application Design Tool
 defining data with, 3-1 to 3-9
 sample session, 3-4 to 3-8
ASCENDING
 sort key, 6-8
Assigning query names
 using the ADT command, 3-2

B

BETWEEN relational operator, 6-4
BUT Boolean operator, 6-5

C

Collections, 6-10 to 6-21
 changing content of, 6-19 to 6-20
 displaying, 6-11 to 6-12
 erasing, 8-2
 forming, 6-11

Collections (Cont.)
 making new CURRENT from old, 6-14
 to 6-15
 naming, 6-15 to 6-17
 releasing, 6-20 to 6-21
 renaming, 6-18 to 6-19
 SHOW command, 6-14
 showing names of, 6-17 to 6-18
 sorting, 6-12 to 6-14
CONTAINING relational operator, 6-4
CTRL/Z, 2-5, 5-3
CURRENT
 in record selection expressions, 6-15

D

Data dictionary
 creating, 3-9
 placing definitions into, 3-8 to 3-9
Data types, 3-2 to 3-3
DEFINE DICTIONARY command, 3-9
DEFINE DOMAIN command, 3-9
DEFINE FILE command, 3-9
DEFINE RECORD command, 3-9
Defining
 data outside the Application Design
 Tool, 3-9 to 3-10
 data with Application Design Tool, 3-1
 to 3-9
DESCENDING
 sort key, 6-8

Displaying
 DATATRIEVE objects, 2-4
 See also SHOW command
 dictionary name, 2-2

Domains
 planning, 3-2 to 3-4
 sample, 1-3

E

Ending a DATATRIEVE session, 2-5
EQUAL relational operator, 6-4
Erasing data, 8-1 to 8-3
EXIT command, 2-5
Exiting
 an ADT session, 3-5
 DATATRIEVE, 2-5

F

Fields
 data types, 3-2 to 3-3
 naming
 using the ADT command, 3-2
 query names, 3-2
 size, 3-3

Files
 indexed, 3-3 to 3-4, 7-7
 sequential, 3-3 to 3-4

FIND statement, 6-11
FINISH command, 6-21

G

GREATER_EQUAL relational operator,
 6-4
GREATER_THAN relational operator,
 6-4
Guide Mode, 4-3

H

HELP command, 4-1 to 4-2

I

Indexed files, 3-3 to 3-4
 ERASE statement with, 8-1 to 8-3
 modifying, 7-6 to 7-8

Information management, 1-1

Invoking
 an ADT session, 3-6
 DATATRIEVE, 2-1

K

Keys
 alternate index, 3-4
 duplicate values, 3-4
 primary index, 3-3

L

LESS_EQUAL relational operator, 6-4
LESS_THAN relational operator, 6-4

M

MODIFY statement
 ALL, 7-8 to 7-9
 preventing, 7-5 to 7-6
Modifying
 data, 7-1 to 7-10
 selected record, 7-4 to 7-6

N

Naming fields
 using the ADT command, 3-2
NOT Boolean operator, 6-5
NOT_EQUAL relational operator, 6-4

O

OR Boolean operator, 6-5

P

Primary index key, 3-3
PRINT statement, 6-1
 creating reports with, 9-1
 displaying data with, 6-5 to 6-10

Q

Query names, 3-2

R

- READY command, 2-4 to 2-5
- Record selection expression, 6-2 to 6-5
- Record stream, 6-1
 - displaying fields, 6-9 to 6-10
 - erasing, 8-2, 8-3
 - forming, 6-2 to 6-5
 - limiting, 6-2 to 6-6
 - modifying fields, 7-9 to 7-10
 - modifying records in, 7-9 to 7-10
 - sorting, 6-6 to 6-9
 - stopping display, 6-10
- Records
 - with missing values, 5-3
- Relational operators, 6-4t
- RELEASE command, 6-20
- Reports, 9-1 to 9-6
 - general features, 9-1 to 9-2
 - output to disk file, 9-5
 - output to line printer, 9-5
 - print list options, 9-5
 - saving specifications, 9-6
 - specification, 9-2 to 9-5
- Retrieving data, 6-1 to 6-21
 - displaying all records, 6-1
- RSE
 - See Record selection expression*

S

- SELECT statement, 7-1 to 7-5
 - forms of, 7-2 to 7-3
 - SHOW command with, 7-3 to 7-4

- Sequential files, 3-3 to 3-4
 - MODIFY statement with, 8-3
 - modifying, 7-1 to 7-6
- SET GUIDE command, 4-3
- SET statement (Report Writer)
 - COLUMNS_PAGE, 9-3
- SETUP.DTR command
 - RSTS/E systems, 2-2
 - RSX systems, 2-2
- SHOW ALL command, 2-3
- SHOW DICTIONARY command, 2-2
- SHOW DOMAINS command, 2-4
- SHOW READY command, 2-4
- SHOW RECORDS command, 2-4
- Sort keys, 6-7
- Starting
 - an ADT session, 3-6
 - DATATRIEVE, 2-1 to 2-4
- STORE statement, 5-1 to 5-5
 - cancelling, 5-3
 - data types, 5-4
 - REPEAT with, 5-3

T

- Terminology
 - DATATRIEVE, 1-2

HOW TO ORDER ADDITIONAL DOCUMENTATION

DIRECT TELEPHONE ORDERS

In Continental USA
and Puerto Rico
call **800-258-1710**

In Canada
call **800-267-6146**

In New Hampshire,
Alaska or Hawaii
call **603-884-6660**

DIRECT MAIL ORDERS (U.S. and Puerto Rico*)

DIGITAL EQUIPMENT CORPORATION
P.O. Box CS2008
Nashua, New Hampshire 03061

DIRECT MAIL ORDERS (Canada)

DIGITAL EQUIPMENT OF CANADA LTD.
940 Belfast Road
Ottawa, Ontario, Canada K1G 4C2
Attn: A&SG Business Manager

INTERNATIONAL

DIGITAL EQUIPMENT CORPORATION
A&SG Business Manager
c/o Digital's local subsidiary
or approved distributor

Internal orders should be placed through the Software Distribution Center (SDC), Digital Equipment Corporation, Northboro, Massachusetts 01532

*Any prepaid order from Puerto Rico must be placed
with the Local Digital Subsidiary:
809-754-7575

Reader's Comments

Note: This form is for document comments only. DIGITAL will use comments submitted on this form at the company's discretion. If you require a written reply and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Did you find this manual understandable, usable, and well-organized? Please make suggestions for improvement. _____

Did you find errors in this manual? If so, specify the error and the page number. _____

Please indicate the type of user/reader that you most nearly represent.

- Assembly language programmer
- Higher-level language programmer
- Occasional programmer (experienced)
- User with little programming experience
- Student programmer
- Other (please specify) _____

Name _____ Date _____

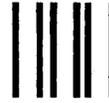
Organization _____

Street _____

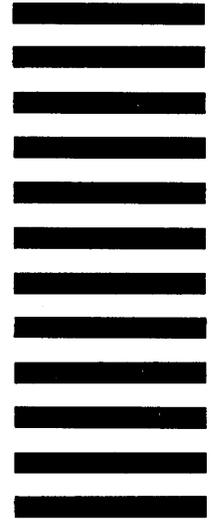
City _____ State _____ Zip Code
or
Country _____

-----Do Not Tear - Fold Here and Tape-----

digital



No Postage
Necessary
if Mailed in the
United States



BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO.33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

ATTN: DISG Documentation ZK02-2/N53
DIGITAL EQUIPMENT CORPORATION
110 SPIT BROOK ROAD
NASHUA, N.H. 03062

-----Do Not Tear - Fold Here and Tape-----

Update Notice #1

November 1987

Introduction to DATATRIEVE-11

AD-X025A-T1

Copyright © 1987 by Digital Equipment Corporation.
All Rights Reserved.

New and Changed Information

This update contains changes and additions made to the
Introduction to DATATRIEVE-11 for Version 3.2.

Instructions

Place the enclosed pages in the *Introduction to DATATRIEVE-11*
Version 3 as replacements for or additions to current pages. Change
bars on replacement pages indicate changed text. For new pages and
pages where most of the text has been substantially revised, no
change bars are used. Instead, only the Version 3.2 release date is
shown on the bottom corner of the page.

Old Page

Title Page/Copyright
iii to x
2-1/2-2
Chapter 4
Chapter 5
Index
How to Order
Reader's Comments/Mailer

New Page

Title Page/Copyright
iii to viii
2-1/2-2
Chapter 4
Chapter 5
Index
How to Order
Reader's Comments/Mailers

Introduction to DATATRIEVE-11

Order No. AA-X025A-TK
Including AD-X025A-T1

November 1987

This document introduces you to basic DATATRIEVE-11 commands.

OPERATING SYSTEMS: RSX-11M
RSX11M-PLUS
RSTS/E
Micro/RSX
Micro/RSTS
VMS with RSX-11

SOFTWARE VERSION: DATATRIEVE-11 V3.2

digital equipment corporation, maynard, massachusetts

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by DIGITAL or its affiliated companies.

Copyright © 1983, 1987 by Digital Equipment Corporation. All Rights Reserved.

The postage-paid Reader's Comments forms at the end of this document request your critical evaluation to assist us in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

digital™

ACMS

CDD

DATATRIEVE

DEC

DECnet

DECUS

Micro/RSTS

Micro/RSX

MicroVAX

MicroVMS

PDP

RALLY

Rdb/ELN

Rdb/VMS

ReGIS

RSTS

RSX

TDMS

TEAMDATA

UNIBUS

VAX

VAXcluster

VAXinfo

VAX Information Architecture

VAX/VMS

VMS

VT

Contents

	Page
How to Use This Manual	vii
1 Introduction	
1.1 DATATRIEVE Terminology	1-2
1.2 Using DATATRIEVE-11.	1-2
1.3 The Sample Domains	1-2
2 Starting and Stopping a DATATRIEVE Session	
2.1 Starting DATATRIEVE	2-1
2.2 Displaying Information About Your DATATRIEVE Environment	2-4
2.3 Accessing Domains	2-4
2.4 Ending a DATATRIEVE Session	2-5
3 Defining Data with ADT	
3.1 Planning a DATATRIEVE Domain	3-2
3.2 Using ADT	3-4
3.2.1 Invoking the .CMD File	3-8
3.3 Defining Domains Outside ADT	3-9
4 HELP and GUIDE Mode: Quick-Reference Section	
4.1 Using DATATRIEVE HELP.	4-1
4.2 Using GUIDE Mode	4-2
5 Storing Data	
5.1 Preparing a Domain for Storing Data	5-1
5.2 Using the STORE Statement	5-2
5.3 Preventing a STORE Statement from Taking Effect	5-2
5.4 Creating a New Record Without Complete Information	5-3
5.5 Storing Two or More Records At a Time	5-3
5.6 Storing the Correct Type of Data in New Records	5-4

6 Retrieving Data

6.1	Displaying All Records in a Domain	6-1
6.2	Selecting a Group of Records	6-2
6.2.1	Specifying the Number of Records in a Record Stream	6-2
6.2.2	Limiting Record Streams by Specifying Values	6-3
6.3	Examples of Displaying Data with the PRINT Statement	6-5
6.4	Sorting Records in Record Streams	6-7
6.5	Displaying Fields from Record Streams	6-9
6.6	Stopping the Display of Records	6-10
6.7	Using Collections	6-11
6.7.1	Forming Collections	6-11
6.7.2	Displaying the CURRENT Collection	6-11
6.7.3	Displaying Sorted Records in the CURRENT Collection	6-13
6.7.4	Displaying Information About the CURRENT Collection	6-14
6.7.5	Making a New CURRENT Collection from an Old One	6-14
6.7.6	Using the CURRENT Collection in RSEs	6-15
6.8	Naming a Collection.	6-15
6.8.1	Showing the Names of Existing Collections	6-17
6.8.2	Naming an Unnamed Collection.	6-18
6.8.3	Changing the Content of a Named Collection	6-19
6.9	Releasing Collections	6-20
6.10	Using the FINISH Command	6-21

7 Modifying Data

7.1	Using the SELECT Statement with a Collection	7-1
7.2	Using Other Forms of the SELECT Statement.	7-2
7.3	Using the SHOW Command to Identify the Selected Record	7-4
7.4	Modifying a Selected Record in a Collection	7-4
7.5	Modifying Fields in Indexed Data Files	7-6
7.6	Modifying All Records in a Collection	7-8
7.7	Modifying Records in a Record Stream.	7-9
7.8	Modifying Specific Fields in a Record Stream	7-10

8 Erasing Data

8.1	Using the ERASE Statement on Indexed Files.	8-1
8.1.1	Erasing a Selected Record	8-1
8.1.2	Erasing a Collection of Records	8-2
8.1.3	Erasing a Record Stream	8-3
8.2	Using the MODIFY Statement on Sequential Files	8-3

9 Writing Simple Reports

9.1	The General Features of a Report	9-1
9.2	The Report Specification.	9-2
9.2.1	Multiple-Line Report Names.	9-3
9.2.2	The Columns_Page Option	9-3
9.2.3	Print List Options	9-5
9.3	Output of Reports	9-5
9.3.1	Output to a Line Printer	9-5
9.3.2	Output to a Disk File	9-5
9.4	Saving Report Specifications.	9-6
9.5	Summary	9-6

Glossary

Index

Figures

1-1	The DATATRIEVE Domain	1-3
4-1	The HELP Screen	4-1
4-2	The HELP SHOW screen	4-2

Tables

6-1	Relational Operators	6-4
-----	--------------------------------	-----

How to Use This Manual

This book introduces you to DATATRIEVE-11. It begins with a brief explanation of information management and the use of DATATRIEVE. It then guides you through the basic tasks you can perform with DATATRIEVE.

Intended Audience

The *Introduction to DATATRIEVE-11* assumes you can use your terminal to log in to your local computer system and display the contents of your file directory. If you need to review how to do that, consult the user's guide for your operating system.

This book addresses people who are unfamiliar with DATATRIEVE-11.

Structure

There are nine chapters in this introduction:

- Chapter 1 Briefly describes information management and lists some basic DATATRIEVE functions
- Chapter 2 Supplies information about what you have to do to begin using DATATRIEVE, describes some of the basic commands and statements that tell DATATRIEVE what to do, and tells you how to leave DATATRIEVE
- Chapter 3 Explains how to define information for DATATRIEVE using ADT
- Chapter 4 Describes the use of Help and Guide Mode to give you online assistance while using DATATRIEVE
- Chapter 5 Explains how to store information with DATATRIEVE
- Chapter 6 Describes a variety of ways you can retrieve data

- Chapter 7 Contains information about modifying information you have stored previously
- Chapter 8 Explains how to erase data
- Chapter 9 Describes some of the basic features of the DATATRIEVE Report Writer

Conventions

The following list explains the special symbols used in this book:

RET

This symbol tells you to press the RETURN key on the keyboard of your terminal.

TAB

This symbol tells you to press the TAB key on the keyboard of your terminal.

CTRL/X

This symbol tells you to press the CTRL (control) key and a letter key (usually Z or C) at the same time.

>

The symbol > on a line by itself represents the system prompt.

color

Text printed in red shows the lines you type while doing the examples.

Related Manuals

DATATRIEVE-11 Guide to Writing Reports

DATATRIEVE-11 User's Guide

Starting and Stopping a DATATRIEVE Session **2**

This chapter shows you how to:

- Start DATATRIEVE
- Show what data is available to you
- Get ready to display your data
- Stop DATATRIEVE

2.1 Starting DATATRIEVE

To start DATATRIEVE, type:

```
DTR(RET)
```

DATATRIEVE displays the following message:

```
PDP-11 DATATRIEVE, DEC Query and Report System  
Version: V3.2, 8-Aug-87  
Type HELP for help  
DTR>
```

If you do not get this startup banner, type either RUN \$DTR or the symbol defined by your system manager to invoke DATATRIEVE.

For RSTS/E systems type:

```
DTR> @LB:SETUP.DTR(RET)
```

For RSX systems type:

```
DTR> @LB:[1,2]SETUP.DTR(RET)
```

This command creates a dictionary for you called SAMPLE.DIC, enters the domain and record definitions into your dictionary, and copies the data files into your directory. The command displays the following information:

```
*****  
Datatrieve is now setting up its sample definitions and data in  
your private dictionary SAMPLE.DIC. The data files will be stored  
in your default directory.  
  
There will be 4 data files created:  
YACHT, OWNER, FAMILY and PERSON.  
*****
```

The commands and definitions scroll on your screen, and the display concludes with the message:

```
*****  
The Datatrieve sample definitions and data have been installed.  
*****  
DTR>
```

If you do not know whether you are using a RSTS/E or a RSX system, ask your system manager.

In response to the @SETUP.DTR command, DATATRIEVE may display a message that SAMPLE.DIC already exists. You can ignore this message.

To display the name of your dictionary, type:

```
DTR> SHOW DICTIONARY(RET)
```

For RSTS/E systems DATATRIEVE responds with a message resembling the following:

```
The current dictionary is SY:[1,170]SAMPLE.DIC  
DTR>
```

The message for RSX systems resembles the following:

```
The current dictionary is DB0:[1,2]SAMPLE.DIC  
DTR>
```

HELP and Guide Mode: Quick-Reference Section

4

4.1 Using DATATRIEVE HELP

A HELP program is available that can answer many questions you may have as you are using DATATRIEVE. You can call it by typing HELP and pressing RETURN whenever you are at the DTR> prompt.

When you request HELP, your screen appears as shown in Figure 4-1.

```
DTR> HELP(RET)
Two levels of assistance are available, basic and advanced.
To get help for the elementary DATATRIEVE statements, type
HELP followed by the names of the statements for which help
is required. To get a list of the topics for which help is
available, type
```

```
HELP HELP
```

```
To get assistance with advanced DATATRIEVE statements, type
```

```
HELP ADVANCED
```

```
followed by the names of the statements for which help is
required. To get a list of the topics for which advanced
is available, type
```

```
HELP ADVANCED HELP
```

```
Paired square brackets indicate that the enclosed clause is
optional. Capital letters indicate DATATRIEVE keywords.
Words in lower case indicate that a user supplied name or
value is required.
```

```
DTR>
```

Figure 4-1: The HELP Screen

The best way to become familiar with HELP may be to type HELP HELP at the DTR> prompt. DATATRIEVE displays a list of topics; HELP provides information about each of them. Type HELP and one of those topics to find out more about the topic. While you are learning how to use HELP, you can also learn something about elements of DATATRIEVE.

After you become familiar with the HELP program, type HELP at the DTR>prompt anytime during your session followed by the name of a topic. DATATRIEVE displays the appropriate text on your screen.

Figure 4-2 shows the output when you use HELP to find out more about the keyword SHOW.

```
DTR> HELP SHOW(RET)
The SHOW statement is used to display the definitions of
domains, records, procedures, and tables stored in the data
dictionary, and information concerning the state of various
objects within Datatrieve. The form of the statement is:

        SHOW object-1, object-2, object-3...

where each "object" is one of the following:
    SETS
    DOMAINS
    RECORDS
    PROCEDURES
    TABLES
    COLLECTIONS
    FIELDS
    READY
    CURRENT
    ALL
    DICTIONARY
    SPACE
    domain-name
    record-name
    procedure-name
    table-name
    collection-name
DTR>
```

Figure 4-2: The HELP SHOW screen

4.2 Using Guide Mode

DATATRIEVE offers another type of help in Guide mode. Guide mode is useful when you are just learning DATATRIEVE, because it prompts you with a choice of entries that are correct.

Guide mode works only on DIGITAL video display terminals. If you have a hardcopy terminal and enter a request for Guide mode, DATATRIEVE displays an error message on your terminal and returns you to DATATRIEVE command level (indicated by the DTR> prompt).

To invoke Guide mode, enter:

```
DTR> SET GUIDE(RET)
```

The best way to see how Guide mode works is to use it. Guide mode completes keywords after you have typed only enough letters to uniquely identify them.

For example, if you want to enter the keyword **READY**, you can type **R**, press the space bar, and Guide mode completes the word for you. If you type **RE**, **REA**, and so on, **DATATRIEVE** also completes the word as **READY**.

Be careful that the second letter of the command does not duplicate the first letter of your next word, however. Suppose you want to **READY** the domain **EMPLOYEES**. If you type **R**, a space, then **E**, Guide mode reads the **E** as the second letter in **READY** instead of the first **E** in **EMPLOYEES**. You must type **RE**, a space, and then the **E** of **EMPLOYEES**.

When you enter **SET GUIDE** and press **RETURN**, **DATATRIEVE** clears the terminal screen and prompts you to enter a command or type a question mark (?) for help. If you request help, **DATATRIEVE** displays all the possible words you can currently use.

Throughout, Guide mode prompts you for the next part of your entry, including spaces, quotation marks, and carriage returns. Not all options are acceptable in Guide mode, but you can find out which ones are at any time by typing a question mark (?), and **DATATRIEVE** displays all the acceptable alternatives. **DATATRIEVE** displays an error message if the entry you try is not allowed.

You can stop Guide mode by typing **LEAVE**. **DATATRIEVE** then displays a message on your terminal and returns you to **DATATRIEVE** command level (indicated by the **DTR>** prompt):

```
LEAVE(RET)
```

```
You're on your own now. Good luck!
```

```
DTR>
```

Although Guide mode is primarily a training tool, you may also find it useful to save keystrokes while typing. Experiment with it, and see the results you get. You can then judge how to use Guide mode to your advantage.

Storing Data 5

This chapter shows you how to add new records to a data file using the STORE statement.

5.1 Preparing a Domain for Storing Data

To store new records in a data file, you must ready the domain for WRITE or EXTEND access. With the WRITE command you can store, read, erase, and change records. With the EXTEND command you can only store records. The examples in this chapter use the PERSONNEL domain readied for WRITE access:

```
DTR> READY PERSONNEL WRITE(RET)
DTR>
```

The DTR> prompt indicates that DATATRIEVE has readied the domain. Verify the type of access by entering:

```
DTR> SHOW READY(RET)
Ready domains:

                PERSONNEL: RMS INDEXED, PROTECTED WRITE

DTR>
```

The phrase PROTECTED WRITE confirms that you have write access to the PERSONNEL domain. You can now store records with the STORE statement.

5.2 Using the STORE Statement

To add new records to a data file, use the STORE statement. DATATRIEVE immediately prompts you for the value of each elementary field in the new record. To store a new record in the PERSONNEL domain, type:

```
DTR> STORE PERSONNEL(RET)
Enter ID: 87422(RET)
Enter EMPLOYEE_STATUS: EXPERIENCED(RET)
Enter FIRST_NAME: MARTY(RET)
Enter LAST_NAME: KELLEY(RET)
Enter DEPT: T32(RET)
Enter START_DATE: 26-AUG-1986(RET)
Enter SALARY: 18750(RET)
Enter SUP_ID: 87289(RET)
DTR>
```

Storing a record gives you an opportunity to observe several DATATRIEVE features you have not encountered before.

Because DATATRIEVE is case sensitive in certain circumstances, you should make a habit of using all uppercase letters.

Note that DATATRIEVE displays a date in the format you specify in the record definition, not in the format you use when you store the date. In the previous example you entered the date as 26-AUG-1986. You could also have entered it as 08-26-86. In either case, DATATRIEVE displays the date as 26-Aug-1986.

Similarly, the record definition determines the format for the SALARY field. You enter the salary without a dollar sign. The record definition determines the form the field takes as output. The PERSONNEL_REC specifies that the dollar sign be printed.

If you did the preceding example using your own PERSONNEL database, you actually added this employee to your file.

5.3 Preventing a STORE Statement from Taking Effect

If you start to enter a STORE statement but then decide you want to cancel it, press CTRL/Z at any point before you finish creating or modifying the record.

```
DTR> STORE PERSONNEL(RET)
Enter ID: 07391(RET)
Enter EMPLOYEE_STATUS: TRAINEE (RET)
Enter FIRST_NAME: CHARLEY(RET)
Enter LAST_NAME: ^Z
Execution terminated by operator
DTR>
```

5.4 Creating a New Record Without Complete Information

You can create a new record, even though all the necessary information is not yet available to you. In the following example, a person has been hired but her starting date, salary, and assignment are unknown. If you press a space or TAB key before pressing RETURN, DATATRIEVE sets the field to blank or zero, depending on the field definition. If you press RETURN without making any entry, DATATRIEVE continues to prompt you until you enter something.

```
DTR> STORE PERSONNEL(RET)
Enter ID: (RET)
Enter ID: 12345(RET)
Enter EMPLOYEE_STATUS: TRAINEE(RET)
Enter FIRST_NAME: HENRIETTA(RET)
Enter LAST_NAME: MILLER(RET)
Enter DEPT: (TAB)(RET)
Enter START_DATE: (TAB)(RET)
Enter SALARY: (TAB)(RET)
Enter SUP_ID: (TAB)(RET)
DTR>
```

If you did this example, the entry is now included in your PERSONNEL database, with blanks in the DEPT, START_DATE, and SALARY fields, and 00000 in the SUP_ID field. The PERSONNEL_REC record definition sets all these defaults.

5.5 Storing Two or More Records at a Time

You can store more than one record at a time by using the REPEAT statement. Type REPEAT, the number of records you want to STORE, and the statement STORE PERSONNEL. DATATRIEVE prompts you for the values of each elementary field in each new record:

```
DTR> REPEAT 2 STORE PERSONNEL(RET)
Enter ID: 75639(RET)
Enter EMPLOYEE_STATUS: TRAINEE(RET)
Enter FIRST_NAME: MIKE(RET)
Enter LAST_NAME: MILLER(RET)
Enter DEPT: T44(RET)
Enter START_DATE: 6-FEB-1981(RET)
Enter SALARY: 16750(RET)
Enter SUP_ID: 00012(RET)
Enter ID: 75640(RET)
Enter STATUS: TRAINEE(RET)
Enter FIRST_NAME: BEN(RET)
Enter LAST_NAME: PUTNAM(RET)
Enter DEPT: T44(RET)
Enter START_DATE: 6-FEB-1981(RET)
Enter SALARY: 16500(RET)
Enter SUP_ID: 00012(RET)
DTR>
```

Again, if you did this example, you have added these two employees to your PERSONNEL database.

When you are storing more than one record with a `STORE` command, each single record is stored as you complete the information for it and press the last `RETURN` before continuing with the next record. If you have requested `DATATRIEVE` to store a certain number of records and it turns out that you want to enter fewer than you had planned, you can stop the prompting for more entries with `CTRL/Z`. If you use `CTRL/Z` to terminate the entry of a record, it affects only that record.

5.6 Storing the Correct Type of Data in New Records

The record definition specifies the type of data that you can store in each field of the record. When you create a new record, you must put the correct type of data in each of the fields. For instance, if the record definition requires a number, `DATATRIEVE` refuses to accept letters. If you try to enter the wrong type of data in a field, `DATATRIEVE` prompts you again for valid data.

Look at the `PERSONNEL_REC` displayed in Section 3.3. To the right of each field name the record contains information describing that field. The `PIC IS` clause tells you the field's data type and its length. `Xs` indicate character data, and `9s` indicate number data. The number in parentheses tells you the length of the field. For example, in the field `SUP_ID`, the `PIC IS` clause tells you that you can enter up to five numbers for `SUP_ID`. Sometimes the length of the field is described as a series of `9s` or `Xs`, as in the `DEPT` field. The three `Xs` tell you that `DEPT` contains three characters.

A summary of the type of data for the `PERSONNEL_REC` follows:

- `EMPLOYEE_STATUS`, `FIRST_NAME`, `LAST_NAME`, and `DEPT` accept only character data.
- `START_DATE` accepts only a date.
- `SALARY` and `SUP_ID` accept only numbers.

Sometimes records contain `VALID IF` clauses. For example, notice the clause `VALID IF STATUS EQ "TRAINEE", "EXPERIENCED"` in the field `EMPLOYEE_STATUS`. `EQ` is an expression meaning equal. The `VALID IF` clause specifies the choices that you can use for a particular field. In the `EMPLOYEE_STATUS` field the `VALID IF` clause tells you to enter either `TRAINEE` or `EXPERIENCED` with capital letters when storing information in that field. `DATATRIEVE` does not accept any other entry for the field `EMPLOYEE_STATUS`.

The next example shows the DATATRIEVE response to several of these kinds of entries:

```
DTR> STORE PERSONNEL(RET)
Enter ID: HAROLD(RET)
Non-digit in string "HAROLD", ignoring character(s)
Re-enter ID: 534020(RET)
Truncation during assignment
Re-enter ID: 34020(RET)
Enter EMPLOYEE_STATUS: BEGINNER(RET)
Validation error for EMPLOYEE_STATUS
Re-enter EMPLOYEE_STATUS: TYRO(RET)
Validation error for EMPLOYEE_STATUS
Re-enter EMPLOYEE_STATUS: TRAINEE(RET)
Enter FIRST_NAME: ^Z
Execution terminated by operator
DTR>
```

The next chapter explains how to use the PRINT statement to retrieve data you have stored.

Index

<
See *LESS_THAN* relational operator
>
See *GREATER_THAN* relational operator

A

Accessing domains, 2-4 to 2-5
ADT command, 3-6
 See Also Application Design Tool
AND Boolean operator, 6-5
Application Design Tool
 defining data with, 3-1 to 3-9
 sample session, 3-4 to 3-8
ASCENDING
 sort key, 6-8
Assigning query names
 using the ADT command, 3-2

B

BETWEEN relational operator, 6-4
BUT Boolean operator, 6-5

C

Collections, 6-10 to 6-21
 changing content of, 6-19 to 6-20
 displaying, 6-11 to 6-12
 erasing, 8-2
 forming, 6-11

Collections (Cont.)
 making new CURRENT from old, 6-14 to 6-15
 naming, 6-15 to 6-17
 releasing, 6-20 to 6-21
 renaming, 6-18 to 6-19
 SHOW command, 6-14
 showing names of, 6-17 to 6-18
 sorting, 6-12 to 6-14
CONTAINING relational operator, 6-4
CTRL/Z, 2-5, 5-2
CURRENT
 in record selection expressions, 6-15

D

Data dictionary
 creating, 3-9
 placing definitions into, 3-8 to 3-9
Data types, 3-2 to 3-3
DEFINE DICTIONARY command, 3-9
DEFINE DOMAIN command, 3-9
DEFINE FILE command, 3-9
DEFINE RECORD command, 3-9
Defining
 data outside the Application Design Tool, 3-9 to 3-10
 data with Application Design Tool, 3-1 to 3-9
DESCENDING
 sort key, 6-8

Displaying

- DATATRIEVE objects, 2-4
 - See also SHOW command*
- dictionary name, 2-2

Domains

- planning, 3-2 to 3-4
- sample, 1-3

E

- Ending a DATATRIEVE session, 2-5
- EQUAL relational operator, 6-4
- Erasing data, 8-1 to 8-3
- EXIT command, 2-5
- Exiting
 - an ADT session, 3-5
 - DATATRIEVE, 2-5

F

Fields

- data types, 3-2 to 3-3
- naming
 - using the ADT command, 3-2
- query names, 3-2
- size, 3-3

Files

- indexed, 3-3 to 3-4, 7-7
- sequential, 3-3 to 3-4

- FIND statement, 6-11
- FINISH command, 6-21

G

- GREATER_EQUAL relational operator, 6-4
- GREATER_THAN relational operator, 6-4
- Guide Mode, 4-2, 4-3

H

- HELP command, 4-1, 4-2

I

- Indexed files, 3-3 to 3-4
 - ERASE statement with, 8-1 to 8-3
 - modifying, 7-6 to 7-8

Information management, 1-1

Invoking

- an ADT session, 3-6
- DATATRIEVE, 2-1

K

Keys

- alternate index, 3-4
- duplicate values, 3-4
- primary index, 3-3

L

- LESS_EQUAL relational operator, 6-4
- LESS_THAN relational operator, 6-4

M

MODIFY statement

- ALL, 7-8 to 7-9
- preventing, 7-5 to 7-6

Modifying

- data, 7-1 to 7-10
- selected record, 7-4 to 7-6

N

Naming fields

- using the ADT command, 3-2
- NOT Boolean operator, 6-5
- NOT_EQUAL relational operator, 6-4

O

- OR Boolean operator, 6-5

P

Primary index key, 3-3

PRINT statement, 6-1

- creating reports with, 9-1
- displaying data with, 6-5 to 6-10

Q

- Query names, 3-2

R

- READY command, 2-4 to 2-5
- Record selection expression, 6-2 to 6-5
- Record stream, 6-1
 - displaying fields, 6-9 to 6-10
 - erasing, 8-2, 8-3
 - forming, 6-2 to 6-5
 - limiting, 6-2 to 6-6
 - modifying fields, 7-9 to 7-10
 - modifying records in, 7-9 to 7-10
 - sorting, 6-6 to 6-9
 - stopping display, 6-10
- Records
 - with missing values, 5-3
- Relational operators, 6-4t
- RELEASE command, 6-20
- Reports, 9-1 to 9-6
 - general features, 9-1 to 9-2
 - output to disk file, 9-5
 - output to line printer, 9-5
 - print list options, 9-5
 - saving specifications, 9-6
 - specification, 9-2 to 9-5
- Retrieving data, 6-1 to 6-21
 - displaying all records, 6-1
- RSE
 - See Record selection expression*

S

- SELECT statement, 7-1 to 7-5
 - forms of, 7-2 to 7-3
 - SHOW command with, 7-3 to 7-4

- Sequential files, 3-3 to 3-4
 - MODIFY statement with, 8-3
 - modifying, 7-1 to 7-6
- SET GUIDE command, 4-2
- SET statement (Report Writer)
 - COLUMNS_PAGE, 9-3
- SETUP.DTR command
 - RSTS/E systems, 2-2
 - RSX systems, 2-2
- SHOW ALL command, 2-3
- SHOW DICTIONARY command, 2-2
- SHOW DOMAINS command, 2-4
- SHOW READY command, 2-4
- SHOW RECORDS command, 2-4
- Sort keys, 6-7
- Starting
 - an ADT session, 3-6
 - DATATRIEVE, 2-1 to 2-4
- STORE statement, 5-1
 - cancelling, 5-2
 - data types, 5-4
 - REPEAT with, 5-3

T

- Terminology
 - DATATRIEVE, 1-2

How to Order Additional Documentation

If you live in:	Call:	or Write:
New Hampshire, Alaska	603-884-6660	Digital Equipment Corp. P.O. Box CS2008 Nashua, NH 03061-2698
Continental USA, Puerto Rico, Hawaii	1-800-258-1710	Same as above.
Canada (Ottawa-Hull)	613-234-7726	Digital Equipment Corp. 940 Belfast Road Ottawa, Ontario K1G 4C2 Attn: P&SG Business Manager or approved distributor
Canada (British Columbia)	1-800-267-6146	Same as above.
Canada (All other)	112-800-267-6146	Same as above.
All other areas	—	Digital Equipment Corp. Peripherals & Supplies Centers P&SG Business Manager c/o DIGITAL's local subsidiary

Note: Place prepaid orders from Puerto Rico with the local DIGITAL subsidiary (phone 809-754-7575).

Place internal orders with the Software Distribution Center, Digital Drive, Westminster, MA 01473-0471.

Reader's Comments

Introduction to
DATATRIEVE-11
AA-X025A-TK
Including: AD-X025A-T1

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

I rate this manual's:

	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Completeness (enough information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Clarity (easy to understand)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Organization (structure of subject matter)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Figures (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Examples (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Index (ability to find topic)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Page layout (easy to find information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

I would like to see more/less _____

What I like best about this manual is _____

What I like least about this manual is _____

I found the following errors in this manual:

Page	Description
_____	_____
_____	_____
_____	_____
_____	_____

Additional comments or suggestions to improve this manual:

I am using **Version** _____ of the software this manual describes.

Name/Title _____ Dept. _____

Company _____ Date _____

Mailing Address _____

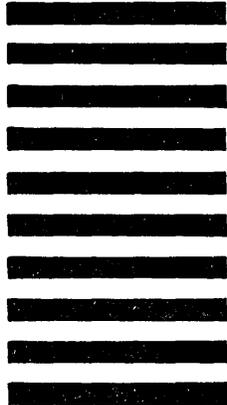
Phone _____

Do Not Tear - Fold Here and Tape

digitalTM



No Postage
Necessary
if Mailed
in the
United States



BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
Corporate User Publications—Spit Brook
ZK01-3/J35
110 SPIT BROOK ROAD
NASHUA, NH 03062-9987



Do Not Tear - Fold Here