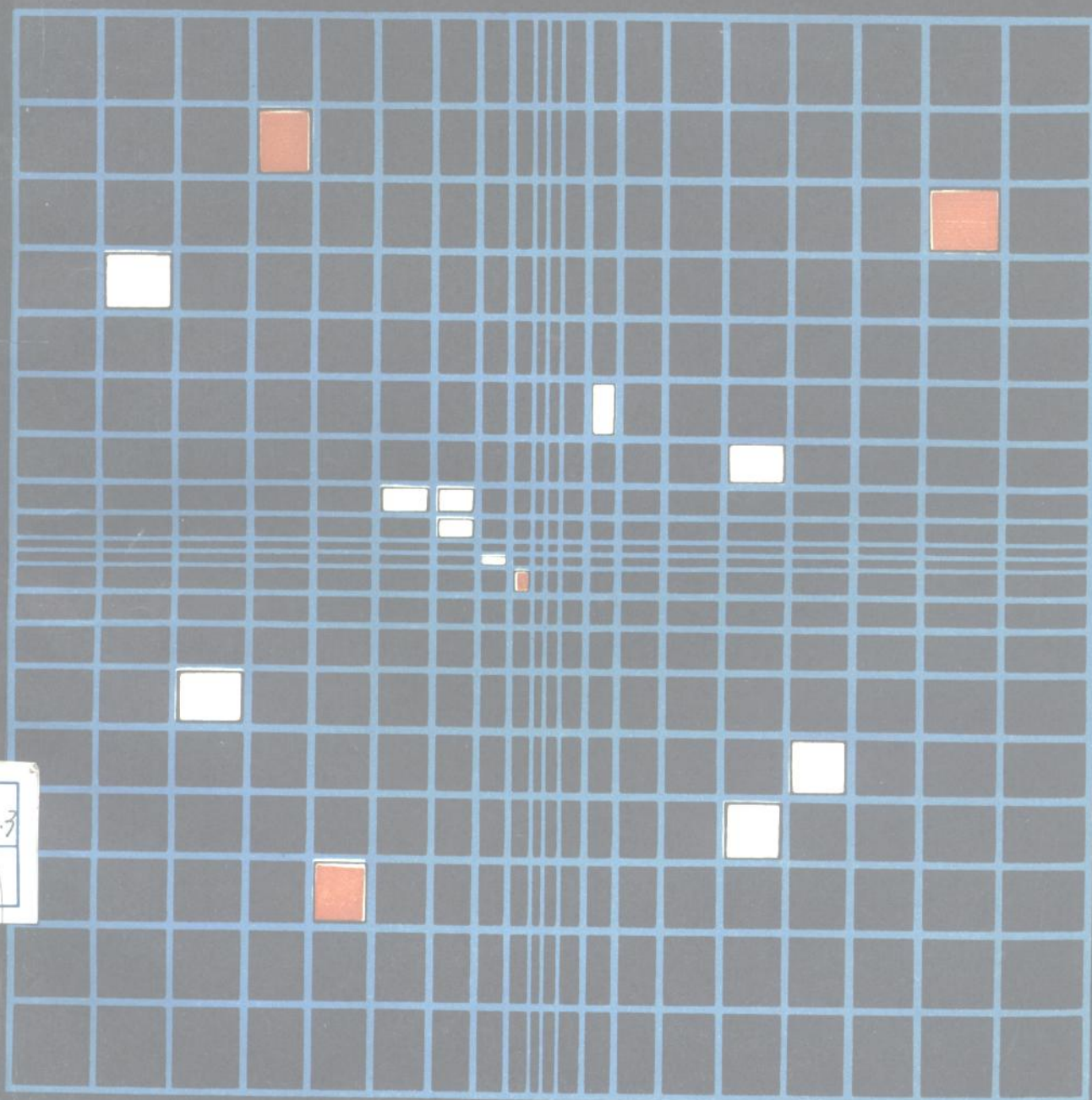


大众数据库

R·A·拜尔斯 著 叶 浩 裴广生 译

上海翻译出版公司



711.132.3

58/1

出版公司

大 众 数 据 库

微型计算机数据库 dBASE-II

叶 澍 裴广生 编译

上 海 翻 译 出 版 公 司

JS320/56
21

大众数据库
微型计算机数据库 dBASE-II

叶 瀚·裘广生 编译

上海翻译出版公司

(上海武定西路1251弄20号)

新华书店上海发行所发行 上海东方印刷厂印刷

开本 787×1092 1/16 印张 8.6 字数 190,000

1986年12月第1版 1986年12月第1次印刷

印数 1—7,000

统一书号: 18811·21 定价: 2.00元

编者的话

近年来,微型电子计算机的应用已深入到社会生活的各个领域。越来越多的人正在使用或期望使用计算机来处理自身的事务。

如何尽快地学习掌握使用电子计算机,已经是广大初学用户面临的十分迫切的问题,也是急需计算机工作者解决的一个重要问题。

一个微型计算机上的数据库管理系统,在这方面提供了极好的机会。它为用户提供了种种工具,用以设计跟用户事务有关的数据库,输入数据库内容,并对之进行查询运算、更新及打印。用户使用数据库技术提供的方便、有效、自然的方法,可以从计算机得到需要的结果。对于初学者,通过学习数据库方法,既可尽快地熟悉计算机(例如,使用本书只需花费20小时左右的时间),又能实际提高他们的工作效率。对于计算机专业人员,使用数据库管理系统进行事务处理,较之使用常规的计算机语言(例如,COBOL、BASIC等)进行同类工作,在程序的编制和调试方面一般可以提高十倍以上的效率。

目前国际上最为流行的微型计算机上的数据库管理系统——dBASE-II,由于它的方便易学、灵活多样等特点,故有“大众数据库”的美称,尤其受到非计算机专业用户的欢迎。

为了向国内广大用户全面介绍这种十分有效的工具,我们主要参考了美国 Robert. A. Byers 所著的“Everyman's Database Primer featuring dBASE-II”一书,编译了本书。叶澍编1~3章,裴广生编4~5章及附录。本书系统地叙述了dBASE-II的基本概念、数据库的设计、数据装填、数据库的更新维护、数据库使用以及过程技巧的应用等。有别于一般的操作手册,本书内容深入浅出、辅以大量实例,可作为数据库技术的入门教材。对于那些不甚了解计算机知识的初学用户尤为适宜。

编者

于上海市计算技术研究所

目 录

编者的话

第一章 数据库常识	1
第一节 什么是数据库	1
第二节 怎样修改数据库	6
第三节 怎样利用数据库	9
第四节 数据库的硬件环境	14
第二章 怎样操纵数据库	17
第一节 数据库设计	17
第二节 数据库的建立	20
第三节 数据库修改和维护	26
第四节 数据库的应用	33
第三章 数据库应用基础	44
第一节 与数据库有关的重要概念	44
第二节 数据库模式	46
第四章 数据库应用技巧	51
第一节 过程的功能	51
第二节 怎样写过程	60
第三节 用于避免错误的过程	75
第四节 用于编制专用报告的过程	79
第五章 数据库应用实例	89
第六章 学习应用数据库点滴体会	103
附录一 术语汇编	104
附录二 dBASE-II 性能指标	117
附录三 dBASE-II 命令分类	121

第一章 数据库常识

第一节 什么是数据库

数据库是计算机行业中的一个专门用语,但它在现实生活中的例子却并不鲜见。譬如,电话号码簿就可以看成是一个小型的“人工”数据库,它含有名字、地址及电话号码。此外,词典、图书编目卡、帐本、职工档案等等,凡是按一定的规则组织在一起的信息集合,都是人工数据库。通常,人们按信息存储的规则查找这类数据库。例如,电话号码簿是按名字(以其笔划或字母序为序)排列的,因而常常按名字为序来查找有关的电话号码或地址。然而,当偶尔需要由地址来查电话号码,甚至由电话号码来查名字时,往往就得翻遍几乎整本电话号码簿,这无异于“大海捞针”了。正是在这种场合,计算机便显露出它超越于人的能力。由于计算机每秒能作几万次比较(这个速度仅指微型计算机而言),因此,如果事先把电话号码簿中的信息存储到计算机中,使之成为一个数据库,那么上述难题便迎刃而解了。计算机用户可从各种角度对数据库提出查询要求,而且几乎总能很快地得到回答。例如,假定一个用户仅知道某人姓王,家住上海徐汇区湖南路30号~100号之内,希望查到他的电话号码。那么,只要提供出上述线索,计算机就能很快地查找到家住湖南路30号~100号的所有姓王者。并把他们的电话号码显示出来。当然此时计算机并不能最终确定用户所要找的究竟是哪个电话号码,但它有助于用户方便而尽快地做到这一点。

图1.1给出了一个电话号码数据库,其中仅包括8个电话号码。真正的数据库中所包含的信息量实际上要多得多,以至于人们不得不借助于计算机。这里我们只不过通过一个简单的例子来说明用户究竟怎样使用一个数据库的。

NAME	ADDRESS	PHONE NUMBER
Byers, Robert A Sr	9999 Glencrest, Standale	555-9242
Byers, Robert A Jr	48 N. Catalina, Pasadena	555-9540
Cassidy, Butch	4800 Rimrock Ct., Sunland	878-1121
Evans, Sydney H.	398 S. Calif. Blvd., Encino	998-1234
Goose, Sil E.	21809 Cottage Ln., Montecito	675-1212
Hedman, Gene	139 Luxury Dr., Bev. Hills	987-6543
Maori, Stanislas	2800 Oak St. #344, Red. Bch.	324-8529
Robertson, James	5892 Glencrest, Standale	997-2741

图1.1 数据库示例

为了建立一个数据库,首先需要有一个数据库管理系统 (DATA BASE MANAGEMENT SYSTEM,简称DBMS)。它为用户提供了种种工具,用以输入数据库的内容,并

对之进行查询、运算、更新及打印。目前在微型计算机上最为流行的 DBMS 名为 dBASE-II。由于它具有方便易学,灵活多样等特点,故有“大众数据库”之美称,尤其受到非计算机专业用户的欢迎。

下面以图 1.1 的数据库为例介绍一下 dBASE-II 的二个基本概念。

记录

记录是组成数据库的基本单位,上述电话号码数据库便是由 8 个记录所组成的,Byers, Robert A Sr 9999 Glencrest, standale 555—9242 就是一个记录。

字段

字段是组成记录的基本单位。上述记录由 3 个字段所组成,它们分别取名为姓名、地址和电话号码。在计算机内则依次写成 NAME、ADDRESS 和 PHONE。

事实上,可以把数据库看成一种表格,其中,行对应于记录,列对应于字段。

现在来看看用户怎样通过 dBASE-II 来建立和使用电话号码数据库。

dBASE-II 的安装和撤消

用户启动微机后,荧光屏上即显示出画面 1.1。



画面 1.1

画面中“CP/M2.2”是目前微机上广为使用的一个操作系统,它使用户能方便地操纵计算机。当然 dBASE-II 也可以在其他操作系统(例如,DOS)支持下运行,但本书中仅以 CP/M2.2 为例。

“A>”是 CP/M 发出的提示符,询问用户要计算机做什么工作。

“◆”是一个光标,指出了用户下一步用键盘键入的内容在画面上的起始位置。

首先,要求 CP/M 把 dBASE-II 安装好,用户键入“DBASE”,如画面1.2所示。



画面 1.2

接着按 RETURN 键(在有些微机上类似的键名为 ENTRY 或 ACCEPT),计算机便接收到用户的命令,随即出现了画面 1.3。

```
CP/M2.2
A>DBASE
PLEASE ENTER TODAY'S DATE MM/DD/YY OR PRESS RETURN;◆
```

画面 1.3

用户键入了当前年、月、日，屏上便显示画面 1.4。

现在 dBASE-II 开始工作 (dBASE-II 有各种不同的版本，本书中假定以 2.3 版为例)。注意，光标左边的“.”是 dBASE-II 的提示符。用户只需在此后键入相应的命令，就可在 dBASE-II 控制下做各种不同的工作。

为了撤消 dBASE-II，让计算机运行其他软件，用户可使用 QUIT 命令。此时计算机将显示出

```
*** END RUN DBASE II ***
```

```
A>
```

于是计算机又返回到 CP/M 控制之下。

```
CP/M2.2
A>DBASE
PLEASE ENTER TODAY'S DATE MM/DD/YY OR PRESS RETURN;
  3/1/82
***DBASEII          VERSION2.3          15FEB1982
◆
```

画面 1.4

数据库的建立

我们接下来具体看看电话号码数据库的建立步骤。首先，由 CREATE 命令对数据库

```
CP/M2.2
A>DBASE
PLEASE ENTER TODAY'S DATE MM/DD/YY OR PRESS RETURN:
  3/1/82
***DBASEII          VERSION2.3          15FEB1982
CREATE
ENTER FILENAME: B:FONEBOOK
ENTER RECORD STRUCTURE AS FOLLOWS:
FIELD          NAME, TYPE, WIDTH, DECIMALPLACES
001            NAME, C, 20
002            ADDRESS, C, 40
003            PHONE, C, 8
004
INPUT DATA NOW? Y◆
```

画面 1.5

进行描述性的定义。这可通过多次键入操作来实现。整个过程如画面 1.5 所示。

首先键入“CREATE”，计算机显示出提示符“ENTER FILENAME”，即要求用户为数据库定义一个名称。在 dBASE-II 中，把表格叫做文件。本例中，文件名称定为“B: FONEBOOK”，意即该文件含义为电话号码簿，存放在计算机 B 驱动器的软盘上。注意，由于一般微机的软盘驱动器分 A、B 二组，故文件名称必须以“A:”或以“B:”起头，其后的字符串由用户根据不同的应用背景任意取定，但最多不能超过 8 个字符，且必须以英文字母为首，中间不能夹空格。

键入文件名称后，下一个提示符为：

ENTER RECORD STRUCTURE AS FOLLOWS,

FIELD NAME, TYPE, WIDTH, DECIMALPLACES

001

这是要求用户对记录的结构进行描述，即定义每个字段的名称、类型、长度及小数点的位置。本例中，定义了三个字段，它们的名称和长度分别为 NAME、20、ADDRESS、40、PHONE、8，它们的类型都是字符型。

在 dBASE-II 中，字段名称由用户任取，但最多不能超过 10 个字符，同样必须以字母起头，且中间不能夹有空格。字段类型共有三种选择，即字符型(C)、数字型(N)和逻辑型(L)。至于字段的长度，由用户根据实际背景而定，比如西方人的姓名，最长不超过 20 个字符，故定义为 20。一个记录中最多允许含有 32 种字段，若超过此限度，可采用第二章中所介绍的一种变通方法来解决。

画面 1.5 中的 001、002、003、004 作为提示符，是每当用户定义一种字段并按 RETURN 键后逐一显示出来的。由于本例中仅含三种字段。因此当显示出 004 后，用户可直接按一下 RETURN 键，于是就显示出

INPUT DATA NOW?

这是计算机问用户是否现在就把现实世界中的数据输入到数据库中去。如果用户暂时不想输入数据，可键入“N”，计算机会自动把前面所做的定义保存起来；否则，用户可键入“Y”。于是计算机将自动把画面 1.5 的内容从荧光屏上清除掉并予以保存，紧接着显出画面 1.6。

RECORD#00001			
NAME	:	◆	:
ADDRESS	:		:
PHONE	:	:	

画面 1.6

至此，第一步工作结束，用户可进入第二步，即输入数据。在画面 1.6 上，沿着“NAME, ◆”键入一个电话持有者的名字时。光标将自动随着向右移动。若该名字正好长度为 20 个字符，则计算机会发出警报(让用户注意不要越界)，光标也将跳到下一个字段 ADDRESS 的最左端，让用户接着键入地址。当然大多数情况下，电话持有者的名字不会长达 20 位，此时，在键入一个名字后，用户必须按 RETURN 键，光标才会跳到下一个字段。当一个记录

RECORD#00001	
NAME	:BYERS, ROBERT A SR:
ADDRESS	:3481 GLENCREST, STANDALE :
PHONE	:555-9242:

画面 1.7

的所有字段的值都被键入后(如画面 1.7 所示),计算机将自动清除原记录 1 画面,并让用户键入记录 2。如此反复,直至全部数据都输入完毕。

前面曾提到,可以把数据库看成一种表格,其中每个记录值对应于一行。这里在数据输入时,一个记录被分成几行来排列。其实这不过是为了操作上的方便起见,一旦数据进入计算机后,同一记录又被自动排在同一行内。因此对应于图 1.5 的含有 8 个记录三种字段的简单数据库,在计算机内将按 8 行 × 3 列来存储。

数据输入完毕后,用户按 RETURN 键,结束前面所发出的 CREATE 命令,此时计算机显示出提示符“.”,等待下一步的命令。

数据库的使用

用户可通过 USE 及 DISPLAY 命令来使用所建立的数据库。首先键入“USE B: FONEBOOK”,通知计算机将要使用的数据库名。一旦计算机再度显示出“.”,便可键入 DISPLAY STRUCTURE 命令,于是计算机显示定义过的有关该数据库结构的全部信息,如画面 1.8 所示。

注意,在文件名称中 FONEBOOK 后增加了“DBF”字样,这是文件的一种类型,在以后各章中还会提到其他种文件类型。画面中的日期是指最后一次数据输入的时间。以后每当数据库中的数据有所更新,日期就会随之而变。另外,这里“TOTAL”后的数字指出了记录的长度(比所有字段长之和多一个字符数)。

如果想查看数据库内所有数据的内容,可使用 DISPLAY ALL 命令。其结果如画面 1.9 所示。

这里最左边一列为记录号码,若用户不希望显示它,可使用 DISPLAY ALL OFF 命

DISPLAY STRUCTURE		(键盘键入的命令)	
STRUCTURE FOR FILE:		B: FONEBOOK. DBF	
NUMBER OF RECORDS:		00008	
DATE OF LAST UPDATE:		9/15/81	
PRIMARY USE DATABASE			
FLD	NAME	TYPE	WIDTH DEC (计算机的响应信息)
001	NAME	C	020
002	ADDRESS	C	040
003	PHONE	C	008
TOTAL		00089	

画面 1.8

.USE B: FONEBOOK			
.DISPLAY ALL			
00001	Byers, Robert A Sr	9999 Glencrest, Standale	555-9242
00002	Byers, Robert A Jr	48 N. Catalina, Pasadena	555-9540
00003	Cassidy, Butch	4800 Rimrock Ct, Sunland	878-1121
00004	Evans, Sydney H.	398 S. Calif. Blvd. Encino	998-1234
00005	Goose, Sil E.	21809 Cottage Ln, Montecito	675-1212
00006	Hedman, Gene	139 Luxury Dr. Bev. Hills	987-6543
00007	Maon, Stanislas	2800 Oak St. #344 Red. Bch.	324-8529
00008	Robertson, James	5892 Glencrest, Standale	997-2741

画面 1.9

令,即结果中将不再包含记录号码。

偶尔,用户需要由电话号码来查找其持有者的名字,此时可用 DISPLAY FOR PHONE = '998-1234',计算机很快就可显示出相应的记录值(此例中,为 00004)。如果在数据库内找不到所键入的号码,则计算机将显示出错误信息。这里电话号码用单引号括起来,它起到了一种分隔符的作用。凡字符型字段的值在命令中都必须用单引号括起来,整个字段值叫做字符串。

前面曾提到计算机能毫不费力地解决诸如由电话持有者的地址或姓名来查找电话号码一类的问题。在 dBASE-II 中,可使用 DISPLAY FOR 'Standale' \$ADDRESS, 来查找家住 Standale 的所有电话持有者。类似地可使用 DISPLAY FOR 'Robert' \$NAME 来查找姓名字段中含有 'Robert' 字样的所有电话持有者。查找结果如画面 1.10 所示。

.DISPLAY FOR 'Robert'\$NAME			
00001	Byers, Robert A Sr	9999 Glencrest, Standale	555-9242
00002	Byers, Robert A Jr	48 N. Catalina, Pasadena	555-9540
00008	Robertson, James	5892 Glencrest, Standale	997-2741

画面 1.10

如果用户仅希望查找其姓氏确切地为 Robert 的电话持有者,则可进一步明确地把命令写成如下形式: DISPLAY FOR, 'Robert' \$NAME,那样,结果中将不包含记录 00008。

综上所述,用户可分三步来建立和使用一个数据库。

- 把数据库看成一种表格,设计其行、列格式,定义文件、字段名及其类型,长度。
- 输入实际数据。
- 此后便可通过 USE、DISPLAY 等命令从各个角度来查看,使用数据库。

第二节 怎样修改数据库

在一个数据库被建立之后,随着时间的推延,必然要求不断修改数据库中原有的内容,以适应现实世界中的变化。例如,随着人们住地的搬迁,其地址和电话号码亦随之而变。此

时就必须相应地更新数据库 FONEBOOK 中有关记录的内容。

记录的删除和增加

记录的删除须分二步进行,先用 DELETE 命令,再用 PACK 命令。画面 1.11 给出了删除名字为“Evans, Sydney H”记录的全过程。

```
.DELETE FOR NAME = 'Evans, Sydney H'
00001 DELETION(S)
.DISPLAY FOR*
00004 *Evans, Sydney H. 398 S. Calif. Blvd., Encino 998.1234
.PACK
PACK COMPLETE 00007 RECORDS COPIED
```

画面 1.11

画面中,执行 DELETE 命令后,欲删除的记录之前被自动地置以记号*,但此时并未真正把该记录从数据库中除掉。DISPLAY FOR * 命令用于显示被删除的记录值,注意此时该记录头部确有记号*。为了真正把该记录从数据库中除掉,可使用 PACK 命令。此后若再显示 FONEBOOK 数据库的内容(见画面 1.12),便可发现只有 7 个记录了,且它们已被重新编号。

00001	Byers, Robert A Sr	9999 Glencrest, Standale	555-9242
00002	Byers, Robert A Jr	48 N. Catalina, Pasadena	555-9540
00003	Cassidy, Butch	4800 Rimrock Ct., Sunland	878-1121
00004	Goose, Sil E.	21809 Cottage Ln., Montecito	675-1212
00005	Hedman, Gene	139 Luxury Dr., Bev. Hills	987-6543
00006	Maori, Stanislas	2800 Oak St. #344, Red. Bch.	324-8529
00007	Robertson, James	5892 Glencrest, Standale	997-2741

画面 1.12

往数据库内增加一个记录只需一步,即使用 APPEND 命令,此时将出现画面 1.13。

于是可键入所要增加的记录内容,其方法与 CREATE 命令相同。假定只知持有者名字,而不知道其地址,则进入 ADDRESS 字段后,可直接按 RETURN 键,以便使光标跳到 PHONE 字段处,详见画面 1.14。

```
RECORD#00008
NAME      : *      :
ADDRESS   :          :
PHONE     :          :
```

画面 1.13

RECORD	00008
NAME	:Deum, T. E.
ADDRESS	:
PHONE	:333-9194:

画面 1.14

字段值的修改

为了修改记录中某些字段的值,可使用 EDIT 和 REPLACE 命令。执行 EDIT 命令时必须同时给出对应记录的序号(在 dBASE-II 中只有少数命令需这样做)。例如,欲修改记录 00008 的地址及电话号码,则可键入 EDIT8。

于是画面 1.14 将再度出现。但此时光标在字段 NAME 处。为了修改地址,必须先设法把它移到字段 ADDRESS 处。注意,在这里不能再靠 RETURN 键来移光标了,而必须使用 CONTROL (或 CTRL)键。

CONTROL 键的作用与 SHIFT 键相类似。

用户用右手按住 CONTROL 键,然后用左手按相应的字母键,便可使光标作出某种移动或引起其他别的作用。表 1.1 给出了若干 CONTROL 键的含义

以下把表 1.1 中的字母键,统称为 CONTROL 键。如“CONTROL X 键”与一般的字母键“X 键”的含义是不同的。

表1.1

CONTROL 键	作 用
X	光标移动到下一个字段
E	光标返回到上一个字段
D	光标向右移一格
S	光标向左移一格
Y	删除字段内容
V	从光标位置起插入字符串
G	删除光标位置上的字符
W	结束 EDIT 命令

上例中,用户可用 CONTROL X 键把光标移到字段 ADDRESS 处,然后键入地址,按 RETURN 键,再键入电话号码,如画面 1.15 所示。

RECORD#	00008
NAME	:Deum, T. E. :
ADDRESS	:111 W. Grandeur, Palos Verdes :
PHONE	:999.3834:

画面 1.15

当把该电话号码的最后一个数字‘4’键入后,计算机会自动结束 EDIT 命令。这是因为这里‘4’恰是数据库中最后一个记录中最后一个字段的最后一个字符。如果不是这种情况,

用户就必须用 CONTROL W 键来通知 dBASE-II, 本次 EDIT 命令已结束, 于是可转而执行其他命令。这里 CONTROL W 的作用类似于写, 即要求 dBASE-II 把修改后的内容记入数据库内原记录位置中。

另一条修改命令 REPLACE 用于按某种准则来修改一个或多个记录中的多个字段值。例如, 欲修改前面键入的最后一个记录的电话号码及地址, 则可键入如下内容:

```
.REPLACE PHONE WITH '222-6661', ADDRESS WITH  
'200 Splended Are, Escondido' FOR NAME = 'Deum, T. E.'
```

计算机将作出如下回答

```
00001 REPLACEMENT (S)
```

意即只有一个记录符合所给定的准则 NAME = 'Deum, T. E.', 且该记录的有关字段值已被修改。

上述这二条修改命令中, EDIT 被称为全画面操作式命令, 即通过画面显示出记录原有的内容, 然后直接键入新的值。而 REPLACE 则是一种自动操作式命令, 用户看不到修改后的新记录, 整个过程由计算机自动完成。

第三节 怎样利用数据库

本节以某零售酒店帐册数据库为例, 来看看 dBASE-II 中另一些常用命令的功能。

假定某零售酒店的帐册含有如下项目:

酒的种类、商标名、规格、现存数量、批发价、零售价。

首先要分别为这些项目定义字段名、类型和长度(见画面 1.16)。其中批发价(COST)及零售价(PRICE)的类型和长度为“N, 6.2”, 意即最大长度为 6 位, 包括 3 位整数、1 个小数点及 2 位小数, 因此其最大值为 999.99。同时为该数据库起一个文件名称: INVENTORY。

画面 1.16、1.17、1.18 给出了该数据库的建立和数据输入过程。

```
.CREATE  
ENTER FILENAME: B: INVENTORY  
ENTER RECORD STRUCTURE AS FOLLOWS:  
FIELD          NAME, TYPE, WIDTH, DECIMAL PLACES  
001             LIQUOR, C, 10  
002             BRAND, C, 20  
003             SIZE, C, 7  
004             QUANTITY, N, 3  
005             COST, N, 6.2  
006             PRICE, N, 6.2  
007  
INPUT DATA NOW? Y
```

画面 1.16

注意,字符型数据输入时从最左空格处开始键入,叫做左对齐。相反,数字型数据则按右对齐原则键入。当误向数字型数据空格内键入字母时,dBASE-II 将不予接受并会鸣响警钟。

LIST 命令

数据库建立后,为了查看全部记录内容,可用 LIST 命令。如画面 1.19 所示。该数据库中共有 15 个记录。

RECORD#00001		
LIQUOR	:	:
BRAND	:	:
SIZE	:	:
QUANTITY	:	:
COST	:	:
PRICE	:	:

画面 1.17

RECORD#00001		
LIQUOR	:	SCOTCH:
BRAND	:	AULD COUNTRY:
SIZE	:	QUART:
QUANTITY	:	23:
COST	:	5.59:
PRICE	:	9.31:

画面 1.18

SUM 命令

在酒店销售管理中通常要了解一些字段值的总计数,此时可利用 SUM 命令。例如,要了解 'SCOTCH' 酒的批发价乘以库存量总数之积,则可键入如下内容:

.SUM COST * QUANTITY FOR LIQUOR = 'SCOTCH'

这里 * 即乘号。计算机通过计算显示出:

438.15

又如要了解规格为 1/4 的 SCOTCH 酒的批发价乘以库存量总数之积则键入内容如下:

.SUM COST * QUANTITY FOR LIQUOR = 'SCOTCH'.AND.SIZE = 'QUART'

128.57

.USE B: INVENTORY				
.LIST				
00001	SCOTCH	AULD COUNTRY	QUART 23	5.59 9.31
00002	SCOTCH	AULD COUNTRY	2LITER 7	9.78 16.30
00003	SCOTCH	AULD COUNTRY	PINT 88	2.74 4.56
00004	VODKA	REAL RUSSIAN	QUART 35	3.78 6.30
00005	VODKA	REAL RUSSIAN	2LITER 9	7.95 13.25
00006	VODKA	REAL RUSSIAN	PINT 75	1.49 2.48
00007	WHISKEY	SOUTHERN RYE	QUART 32	5.11 8.51
00008	WHISKEY	OLD WYOMING	PINT 44	1.98 3.30
00009	WHISKEY	OLD WYOMING	QUART 19	5.29 8.81
00010	WHISKEY	THE NEW SOUTH	QUART 4	7.49 12.48
00011	BOURBON	SOUTHERN ARISTOCRACY	PINT 5	0.99 1.65
00012	BOURBON	SOUTHERN ARISTOCRACY	FIFTH 22	1.78 2.96
00013	BOURBON	SOUTHERN ARISTOCRACY	QUART 21	3.50 5.83
00014	BOURBON	SOUTHERN ARISTOCRACY	1/2 GAL 3	6.89 11.48
00015	BOURBON	SOUTHERN ARISTOCRACY	2LITER 5	6.47 10.78

画面 1.19

REPORT 命令

该命令提供了一种由数据库内数据自动生成报表的功能。在键入 REPORT 命令后, dBASE-II 将会提出一系列问题, 由用户回答, 据此便定义了所需打印报表的格式(参见画面 1.20)。

首先要求在 REPORT FORM NAME 后给出报表名称, 在本例中起名为 B, INVENTORY。报表名称的定义规则与文件名称相同, 但系统将在用户起名后添加 '.FRM' 以区别于文件名称(对于后者, 系统添加 '.DBF')。

接着 dBASE-II 给出了一批有关参数的标准选择, 诸如左边空格数, 每一页的行数、页宽。如用户愿意采用系统内定的标准而不再作任何更动便可按 RETURN 键。

第三步, 显示出 "PAGE HEADING?" 问用户是否需要在每页头上打个页标题。若键入 'Y', 则接着要求用户键入所需打印的页标题内容。本例中为 'LIQUOR STORE INVENTORY'。

第四步, 询问每行之间是否要空一行。

第五步, 询问是否要总计数。若需要, 则再询问是否要分栏小计数; 若需要, 则要求用户键入据以作分栏小计的字段。此例中要求按不同的酒类作小计, 故键入 'LIQUOR', 其结果给出了四种不同的分栏内容(参见图 1.2)。

第六步, 询问是否仅需打印总计报表, 若键入 'N', 则表示要细目报表。本例给出的是一个细目报表。所谓总计报表, 即指仅打印各种不同酒类的小计数, 而不详细列出每个记录。

第七步, 询问当某一分类内容打印完毕后是否要换页。本例并未作出这个要求, 否则将

```

.REPORT
ENTER REPORT FORM NAME: B:INVENTORY
ENTER OPTIONS, M=LEFT MARGIN, L=LINE/PAGE, W=PAGE
WIDTH
PAGE HEADING? (Y/N) Y
ENTER PAGE HEADING: LIQUOR STORE INVENTORY
DOUBLE SPACE REPORT? (Y/N) N
TOTALS REQUIRED IN REPORT? Y
SUBTOTALS IN REPORT? (Y/N) Y
ENTER, SUBTOTALS FIELD: LIQUOR
SUMMARY REPORT, ONLY? (Y/N) N
EJECT PAGE AFTER SUBTOTALS? (Y/N) N
ENTER SUBTOTAL HEADING:
COL    WIDTH, CONTENTS
001    20,BRAND
ENTER HEADING: BRAND
002    7, SIZE
ENTER HEADING: SIZE
003    3, QUANTITY
ENTER HEADING: QTY
ARE SUBTOTALS REQUIRED? (Y/N) Y
004    6, COST
ENTER HEADING: COST
ARE TOTALS REQUIRED? (Y/N) N
005    7, COST*QUANTITY
ENTER HEADING: INVEST
ARE TOTALS REQUIRED? (Y/N) Y
006

```

画面 1.20

把四个小计分别打印在四页上。

第八步,询问是否要打印分栏标题。本例无此需要,故可以按 RETURN 键。

第九步,询问组成报表的各列之宽度、内容及列的标题。所谓报表某列的内容,可以是数据库中的某一字段值,或若干字段的表达式值(例如,“COST * QUANTITY”)。这里第一列列宽为 20,其内容为字段 BRAND 的值,其列标题为‘BRAND’。

上述步骤完成了对报表格式的定义,最后按 RETURN 键,该格式便被保存起来。

此后,用户可随时键入 REPORT 命令,并给出报表名称,dbase-ii 即把数据库的内容按所定义的报表形式显示出来,如图 1.2 所示。若需用打印机印刷,则只须用 REPORT TO PRINT 命令来代替 REPORT 命令。