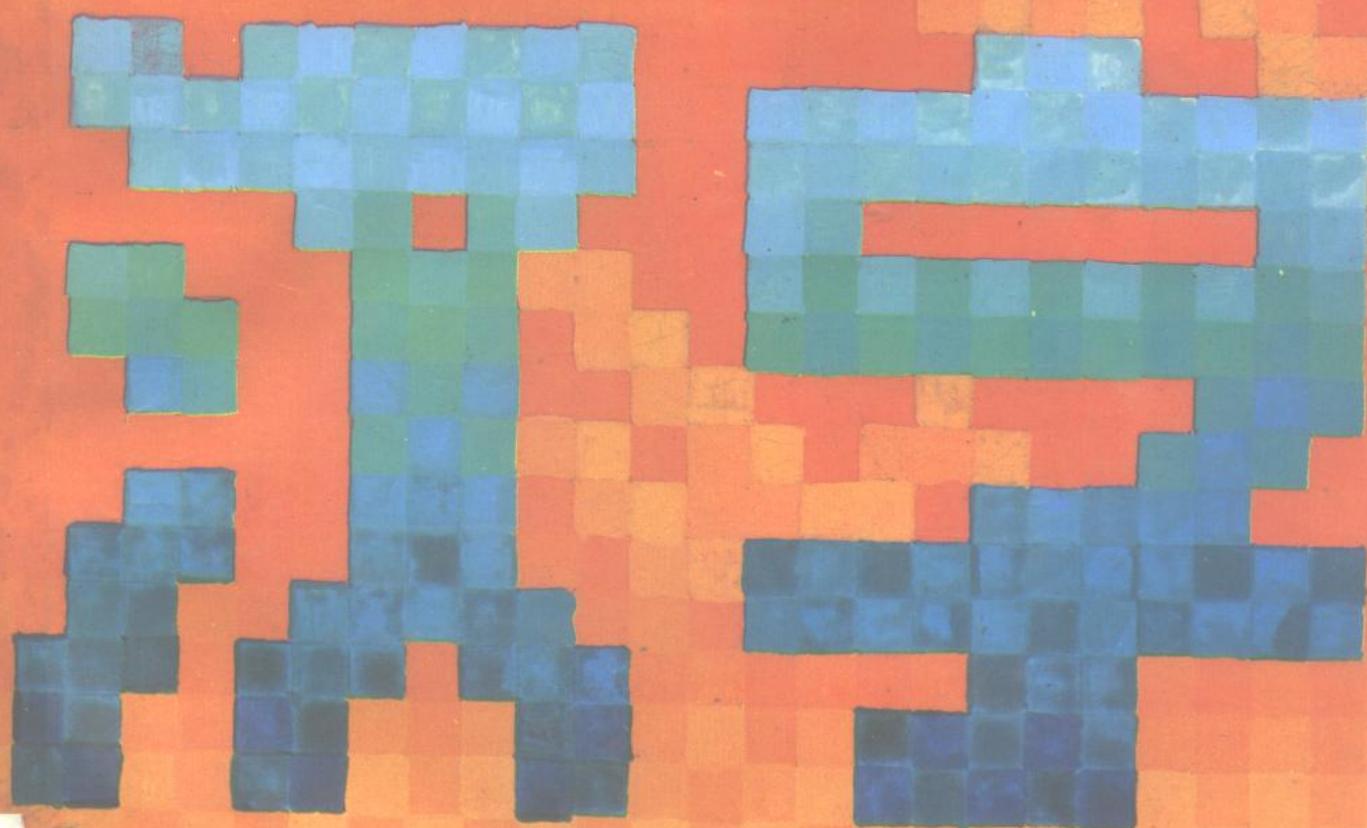


关系数据库 dBASEⅢ 在信息处理中的应用

王国良 主编



兵器工业出版社

关系数据库 dBASEⅢ在 信息处理中的应用

王国良 苑晓青 刘 勇等 编著



兵器工业出版社

1022014

内 容 提 要

本书介绍国内外普遍使用的关系数据库 dBASEⅢ的一般概念、主要功能、命令和函数,报表格式设计工具和多种高级语言之间的数据交换方法,编辑技巧,网络 dBASEⅢPLUS 功能的扩充等,是作者在微机上工作多年的经验总结。本书附有较多的应用实例,可供读者参考选用。

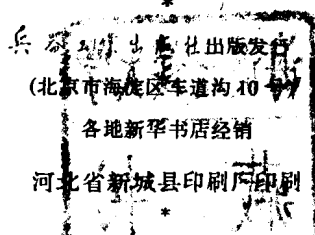
为方便用户,本书选编了 CCDOS 的主要命令,汉字常用输入方法,行编辑程序 EDLIN,字处理 WORDSTAR,动态调试程序 DEBUG 等软件的使用,有了本书,用户可在微机上操作自如。

本书是一本非常实用的参考书,突出编程技巧和应用,可供计算机应用、数据库、信息处理、现代化管理一类的专业人员使用,可作为计算机应用培训班的教材,也可作为工程技术人员、管理人员和广大微机用户的实践指南和自学参考书。

(京)新登字 049 号

关系数据库 dBASEⅢ在信息处理中的应用

王国良 苑晓青 刘 勇等 编著



开本: 787×1092 1/16 印张: 24.625 字数: 605.28 千字

1991年8月第1版 1991年8月第1次印刷

印数: 1—6500 定价: 15.10 元

ISBN 7-80038-337-7/TP·24

前 言

随着计算机科学技术的发展,计算机特别是微型计算机得到了广泛的应用,从科学计算、实时控制逐步扩展到非数值处理的各个领域。特别是在企、事业管理和办公自动化方面,微机显示了他的巨大威力。

人类社会正在进入信息时代,人们在各种活动中都将产生大量的信息。信息需要处理和加工,由于计算机具有高速处理能力和存储容量大的特点,使得人们有可能对大量的信息进行存储和加工处理。数据是信息的具体表现形式,有了数据就产生数据处理的问题。所谓数据处理,它包括对数据的收集、存储、分类、排序、计算或加工、传输、制表等等。信息的处理使有效的资源得到合理的和充分的使用,这种使用促进社会生产力的发展,并又产生出新的信息。

数据处理技术在经历几个发展阶段以后,今天已发展到数据库技术并已得到了广泛的应用。它是实现管理科学化和现代化的强有力的工具。

数据库技术用于大量数据的处理,具有数据结构化,最少冗余度,较高的程序和数据独立性,易于编制应用软件等特点,因而数据库是近年来发展最为迅速的计算机软件。

数据库管理系统是控制和管理计算机的数据资源,使得各种应用程序共享数据资源。

dBASE III数据库管理系统是国际上广为流行的关系型数据库系统。它具有很强的人-机会话式的数据库语言。用dBASE III开发应用软件所需费用要比高级语言低得多,而效率要高的多。dBASE III操作简单,使用方便,尤其在数据库的查询和检索方面,具有较快的速度。

dBASE III在dBASE II的基础上扩展了许多功能。用dBASE II开发的应用软件,可转换成dBASE III的程序。

汉字dBASE III是在英文dBASE III的基础上汉化而成的。它保留了原西文dBASE III的全部功能,是一个中、西文完全兼容的系统。

汉字dBASE III非常适合企、事业管理和办公自动化方面的应用。为进一步推广和扩大它的应用,编者根据几年来的工作实践,编写了这本书,供从事数据处理工作的有关技术人员和管理人员参考,也可作为培训班的教材。

dBASE III PLUS应用日趋广泛,它在dBASE III的基础上又增加了许多功能,特别是它可在网络环境下使用。本书介绍了dBASE III PLUS的功能扩充部份,如命令扩充、函数和网络功能等,以便读者查阅。

本书是在多次培训班教材的基础上,经几次修改整理而成的。

全书共分十五章,第一章到第四章主要介绍dBASE III的主要命令、函数和状态设置,第五章介绍dBASE III报表格式设计工具CDSFORMAT的使用。第六章介绍dBASE III和dBASE II的比较及其转换。第七章介绍dBASE III数据库和高级语言之间的数据交换。第八章介绍dBASE III的编程技巧,第九章介绍增强型dBASE III PLUS功能的扩充。第十章到第十四章为微机使用人员提供磁盘操作系统的主要命令、汉字输入方法、行编辑EDLIN、字处理WORDSTAR和动态调试程序DEBUG这些常用的工具软件,以便随时查阅。第十五

章提供了一些应用实例供读者参考,这些实例已经过了较长时间的实际应用。

书中有些内容参考了有关资料,主要资料目录附后,对这些资料的作者表示感谢。

本书在撰写过程中,得到了机电部五八所领导和许多同志的支持和帮助。兵器工业总公司邵淑贤等同志,为本书的出版给予了热情的鼓励、支持和帮助。本书的责任编辑在审阅加工时提出了不少宝贵意见。不少同志为本书的撰写提供了许多珍贵资料。朱波为本书的输入和校对作了很多工作,在此,谨向这些同志(和为本书印刷出版的同志)表示诚挚的谢意。

本书适合于高等院校计算机专业,特别是有关计算机应用,信息处理,现代化管理一类的专业使用,也可作为计算机应用培训班的教材,并可作为工程技术人员,管理人员和微型计算机用户的实践指南和自学参考书。

数据库管理系统是计算机科学中的一个新兴领域,而且发展又非常迅速,加上我们水平有限,编写时间又很仓促,因此,书中会有不少缺点错误,恳请读者批评指正。

邢进、王晓红、杨锦蓉也参加了部分工作。

编者

1990年12月

目 录

前 言

第一章	dBASEⅢ的一般概念	(1)
§ 1.1	dBASEⅢ的运行环境	(1)
§ 1.2	dBASEⅢ的主要指标	(2)
§ 1.3	数据库的文件类型	(2)
§ 1.4	数据库的结构	(3)
§ 1.5	关于内存变量	(3)
§ 1.6	表达式的概念	(4)
§ 1.7	工作区的选择	(5)
§ 1.8	某些规则	(6)
§ 1.9	符号的约定	(6)
第二章	dBASEⅢ的主要命令	(9)
§ 2.1	数据库的建立	(9)
§ 2.2	数据库文件的打开与关闭	(9)
§ 2.3	在数据库文件中增加记录	(10)
§ 2.4	显示数据库文件的记录和结构	(11)
§ 2.5	定位记录指针	(13)
§ 2.6	记录内容的修改	(14)
§ 2.7	数据库记录的插入和删除	(15)
§ 2.8	数据库记录的整理	(15)
§ 2.9	重新组织数据库	(15)
§ 2.10	内存变量的运算	(19)
§ 2.11	数据库的修改、复制和维护	(24)
§ 2.12	信息的输入和输出	(28)
§ 2.13	命令文件的建立与运行	(33)
第三章	dBASEⅢ的标准函数	(44)
§ 3.1	日期和时间函数	(44)
§ 3.2	字符操作函数	(46)
§ 3.3	数字运算函数	(47)
§ 3.4	测试功能函数	(49)
第四章	dBASEⅢ的 SET 命令	(52)
§ 4.1	控制产生显示信息文件	(52)
§ 4.2	控制铃响	(52)
§ 4.3	复制上一记录的内容	(52)
§ 4.4	控制颜色	(52)

§ 4.5	光标从一个字段到下个字段的移动方式	(53)
§ 4.6	控制显示	(53)
§ 4.7	控制命令行的回复	(53)
§ 4.8	控制回复信息送到打印机	(54)
§ 4.9	控制暂停执行	(54)
§ 4.10	控制显示小数点后位数	(54)
§ 4.11	指定缺省驱动器	(54)
§ 4.12	掩盖或处理有删除标记的记录	(55)
§ 4.13	在全屏幕方式中怎样表示字段宽度	(55)
§ 4.14	@..... SAY 命令的结果显示或打印	(55)
§ 4.15	用 ESC 键终止命令文件的执行	(55)
§ 4.16	两字符串比较是否需要全同	(56)
§ 4.17	数据文件只出现满足指定条件的记录	(56)
§ 4.18	固定小数点后的位数	(56)
§ 4.19	使用由 dFORMAT 程序形成的格式文件	(56)
§ 4.20	重新定义功能键	(57)
§ 4.21	在某些命令的标准显示中是否使用栏标题	(57)
§ 4.22	是否要求帮助性提示	(57)
§ 4.23	打开已存在的索引文件	(58)
§ 4.24	控制反象显示字段	(58)
§ 4.25	设置打印机的左部边界	(58)
§ 4.26	菜单显示控制	(58)
§ 4.27	选定路径	(58)
§ 4.28	输出结果是否送打印机	(59)
§ 4.29	打开一个过程文件	(59)
§ 4.30	连接两个数据库文件	(59)
§ 4.31	防止文件被重写的保护	(60)
§ 4.32	命令执行的结果是否送到屏幕	(60)
§ 4.33	使用带有相同关键字的记录	(60)
第五章	报表格式设计工具 CdFORMAT 的使用	(61)
§ 5.1	CdFORMAT 程序的启动	(61)
§ 5.2	格式文件生成	(62)
§ 5.3	编辑命令	(63)
§ 5.4	某些符号及意义	(65)
§ 5.5	格式文件的调用	(67)
§ 5.6	联机手册使用简述	(68)
第六章	dBASE III 和 dBASE II 的比较	(69)
§ 6.1	技术指标及性能的比较	(69)
§ 6.2	功能比较	(70)
§ 6.3	字段变量、内存变量和空变量的比较	(71)

§ 6.4	函数的比较	(72)
§ 6.5	数据库功能强化综述	(74)
§ 6.6	dBASE II 向 dBASE III 转化	(74)
第七章	dBASE III 数据库和高级语言之间的数据交换	(78)
§ 7.1	概述	(78)
§ 7.2	dBASE III 数据传递命令	(79)
§ 7.3	dBASE III 与 BASIC 语言的数据交换	(82)
§ 7.4	dBASE III 与 FORTRAN 语言的数据交换	(84)
§ 7.5	dBASE III 与 COBOL 语言的数据交换	(86)
§ 7.6	dBASE III 与 PASCAL 语言的数据交换	(89)
§ 7.7	dBASE III 与 C 语言的数据交换	(90)
§ 7.8	dBASE III 与 Supre Calc 3 之间的数据交换	(92)
第八章	dBASE III 的编程技巧	(96)
§ 8.1	如何避免数据的丢失	(96)
§ 8.2	CONFIG.DB 文件的应用	(98)
§ 8.3	如何提高 dBASE III 程序运行速度	(100)
§ 8.4	如何节约程序输入时间	(101)
§ 8.5	dBASE III 如何处理数组	(102)
§ 8.6	dBASE III 多库关联的实现	(103)
§ 8.7	模块公用技巧	(104)
§ 8.8	文件联接速度的提高	(105)
§ 8.9	数值"0"不打印技术	(106)
§ 8.10	多维表格数据库的建立	(107)
§ 8.11	用两次 TOTAL 提高 dBASE 的分类汇总速度	(108)
§ 8.12	用 WAIT 语句设置保密口令	(109)
§ 8.13	随机数的产生	(111)
§ 8.14	dBASE 中使用 TOTAL 命令的一个诀窍	(112)
§ 8.15	在程序中实现屏幕硬拷贝的方法	(112)
§ 8.16	设置 Wordstar 颜色的方法	(113)
§ 8.17	用 BASIC 在 dBASE III 程序中加密的一种方法	(113)
§ 8.18	对文件加密与解密的另一种方法	(114)
§ 8.19	在 dBASE III 中实现数字金额与中文大写转换	(115)
§ 8.20	自动生成数据库结构的说明文件	(117)
第九章	增强型 dBASE III PLUS 功能的扩充	(119)
§ 9.1	dBASE III PLUS 软件结构及特点	(119)
§ 9.2	dBASE III PLUS 的性能指标	(120)
§ 9.3	dBASE III PLUS 功能扩充	(120)
第十章	CCDOS 磁盘操作系统	(133)
§ 10.1	CCDOS 简介	(133)
§ 10.2	基本概念	(133)

§ 10.3	CCDOS 的主要命令	(139)
§ 10.4	批处理命令	(153)
第十一章	汉字的输入方法	(159)
§ 11.1	基本操作	(159)
§ 11.2	汉字输入方法	(161)
§ 11.3	汉字输入中的几点注意事项	(169)
§ 11.4	CTRL+F 功能键的使用	(169)
第十二章	行编辑程序 EDLIN.COM 的使用	(171)
§ 12.1	程序功能	(171)
§ 12.2	启动行编辑程序	(171)
§ 12.3	EDLIN 命令参数	(172)
§ 12.4	插入行命令	(173)
§ 12.5	显示行命令	(175)
§ 12.6	删除行命令	(176)
§ 12.7	编辑行命令	(178)
§ 12.8	替换文字段命令	(178)
§ 12.9	查找文字段命令	(181)
§ 12.10	附加行命令	(182)
§ 12.11	写入行命令	(182)
§ 12.12	结束编辑程序命令	(183)
§ 12.13	退出编辑程序命令	(183)
§ 12.14	复制行命令	(183)
§ 12.15	移动行命令	(184)
§ 12.16	页命令	(185)
§ 12.17	传输行命令	(185)
§ 12.18	命令总结	(185)
第十三章	汉字文字编辑 WORDSTAR 的使用	(187)
§ 13.1	WORDSTAR 的进入和退出	(187)
§ 13.2	文字编辑功能(D 功能)的使用	(187)
§ 13.3	WS 的其他功能	(193)
§ 13.4	编辑命令小结	(195)
第十四章	DEBUG 动态调试程序的使用	(198)
§ 14.1	如何启动 DEBUG 程序	(198)
§ 14.2	DEBUG 命令参数说明	(198)
§ 14.3	DEBUG 命令及其使用	(199)
第十五章	dBASEⅢ的应用实例	(210)
§ 15.1	数据库更新程序	(210)
§ 15.2	屏幕“菜单”生成程序	(212)
§ 15.3	数据库自动连接程序	(216)
§ 15.4	dBASEⅢ的数值处理	(218)

§ 15.5	大容量 dBASEⅢ数据库备份程序	(230)
§ 15.6	绘图程序	(233)
§ 15.7	报表打印程序	(237)
§ 15.8	通用管理程序的设计	(242)
§ 15.9	库房管理系统	(251)
§ 15.10	报刊订阅管理系统	(297)
§ 15.11	设备技术资料借阅程序	(329)
§ 15.12	人事档案管理系统	(337)
附录一	dBASEⅢ命令及功能简述	(365)
附录二	ASCII 码字符集	(381)
参考文献		(383)

第一章 dBASEⅢ的一般概念

世界是一个物质的世界，所有物质形成一个物质流。人们就处在这个物质流中。同时，我们也是生活在一个信息的世界里，所有的信息形成一个信息流。物质的存在伴随着信息存在，物质的变化会引起信息的变化。

今天的人类社会将进入信息化的社会，人们的各种活动都将产生大量的信息，信息需要加工、处理、交流和使用。随着计算机技术的迅速发展，人们有可能对日益增多的大量的信息进行保存和加工处理。

信息的处理经过了手工处理、机械处理和计算机处理三个阶段。当今，用计算机进行信息处理的方法的研究已成为计算机科学技术的主要课题之一。而数据库技术已成为社会信息化时代不可缺少的方法和工具。

数据库系统是在文件管理系统的基础上发展起来的，它解决了文件管理系统数据冗余和数据独立性的问题。其基本思想就是要对所有数据实行统一的、集中的、独立的管理。数据独立于程序而存在，并可以提供给各类不同用户共享使用。

数据的结构一般可分为关系数据结构、层次结构和网络结构三种。因此相应的数据库分为关系数据库、层次数据库和网络数据库。

dBASEⅢ是美国 Ashton-Tate 公司 1984 年 7 月推出的关系数据库，它是在 dBASEⅡ的基础上发展起来的。适用于 IBM PC/XT 及其兼容计算机。dBASEⅢ无论在技术指标方面，还是在系统功能方面都大大优于 dBASEⅡ。dBASEⅢ被引入我国后，对该系统进行了汉化工作，目前我国正得到愈来愈广泛的应用，成为数据处理的有力工具。dBASEⅢ简单易学，具有丰富的提示信息和各种具体的出错指示，利用 dBASEⅢ能够建立完善的数据管理系统，对数据库中的数据，能很方便地进行各种操作。

在 CCDOS 支持下，dBASEⅢ可以处理汉字信息，可实现汉字菜单提示，操作方便灵活，非常适合各类管理人员使用，这就极大地提高了汉字 dBASEⅢ普及推广的应用价值。

§ 1.1 dBASEⅢ的运行环境

dBASEⅢ是由 C 语言编写的，运行时至少占用 180KB 的内存，加上操作系统所需内存，运行 dBASEⅢ至少需要 256KB 的系统存储器。内存大可提高运算速度。

dBASEⅢ可以在 IBM-PC 以及所有 IBM-PC 的兼容机上，在 MS-DOS、PC-DOS 或 CCDOS 操作系统支持下工作。硬件需要二台 360KB 的软盘驱动器或一台 360KB 的软盘驱动器和一台硬盘驱动器，一台宽行打印机。

dBASEⅢ可在软盘上运行，也可在硬盘上运行。若在硬盘上运行 dBASEⅢ，需要将可以拷贝到硬盘上的 dBASEⅢ系统，用 COPY 命令将其拷贝到硬盘 C 中，然后可在硬盘 C 中运行 dBASEⅢ。未经解密的 dBASEⅢ系统主盘，拷贝到硬盘后是不能运行的。

在软盘或硬盘上运行 dBASEⅢ：

A:>[X:] dBASE <CR> (<CR>表示按回车键，下同)

其中: A> 为系统提示符, X: 为驱动器名, 可以缺省。

此时系统进入 dBASEⅢ, 以小圆点·作为 dBASEⅢ的提示符, 这时可执行 dBASE 的各种命令。用 QUIT 命令退出 dBASE, 返回操作系统。

§ 1.2 dBASEⅢ的主要指标

dBASEⅢ可用于交互式运行或应用程序的开发, 具有较完整的关系运算特性, 使用方便。

dBASEⅢ比 dBASEⅡ在功能上有较大的扩充和改进。dBASEⅢ最多同时可打开 15 个文件, 最多同时可打开 10 个数据库文件。一个数据库可以有 10 亿个记录, 每个数据库记录可以有 4000 个字符, 可有 128 个字段, 每个字段最多可有 254 个字符。允许的内存变量最多可有 256 个, 可占 6000 个字节。数值精度为 15.9 位, 符合 IEEE 的浮点标准格式。数值范围为 $1.0E^{-307} \sim 1.0E^{308}$ 。新型的备注字段 MEMO 数据类型可达 50 万个字符。

dBASEⅢ的系统参数, 存储在 CONFIG.DB 文件中, 此文件包含一些初始值, 它大部分是 SET 命令所设置的参数和功能键值, 还有一些提示信息。存储器的大小, 程序的嵌套数等信息可在 CONFIG.DB 中重新定义, 当进入 dBASEⅢ时, 以 CONFIG.DB 中参数值代替缺省值。

§ 1.3 数据库的文件类型

dBASEⅢ允许九种文件类型。数据库文件由文件名和扩展名(也称后缀)组成。文件名必须由字母开头, 可由字母、数字和下划线组成, 最多不超过 8 个字符。空格不能出现在文件名中, 同时, 不允许用单个字母 A, B, C, D, E, F, G, H, I, J 这 10 个字母作为文件名。

文件的扩展名由 1 至 3 个字符组成, 扩展名和文件名之间用小圆点“.”分开。数据库文件的类型由扩展名区分, 共有以下九种类型:

(1) .DBF 为数据库的主文件, 由结构和若干记录组成。

(2) .DBT 为备注文件, 是 .DBF 的辅助文件, 在 .DBT 中引入备注型数据字段 MEMO 时, 相应地就建立了 .DBT 文件, 它最多允许 4000 个字符, 按字处理方式使用。如果字段的内容少于 254 个字符时, 最好不要使用备注型, 因为它降低了运行效率。如不使用 MEMO 字段, 就不产生 .DBT 文件。

(3) .NDX 为索引文件, 是根据关键字表达式在逻辑上进行了排序, 便于检索。使用时索引文件和 .DBF 文件一起打开, 或用 SET 命令来设置打开。

(4) .MEM 为内存变量文件, 由 SAVE TO < 文件名 > 而形成, 它保留在磁盘上, 需要时可再从磁盘上用 RESTORE FROM < 文件名 > 调入。

(5) .PRG 为命令文件, 用文本编辑程序 EDLIN 或修改命令 MODIFY COMMAND 来建立。执行时用 DO < 文件名 >。

(6) .FMT 为格式文件, 它是由报表生成程序 dFORMAT.EXE 来生成(汉字报表生成程序为 dFORMAT.EXE 请参阅第七章)。用 SET FORMAT TO < 文件名.FMT > 和 READ 来使用它。

(7) .FRM 为报告文件, 用 CREAT REPORT 或 MODIFY REPORT 来建立, 由菜单启动, 人机对话方式回答提示信息, 按步建立, 能生成报表, 求小计和总计等。

(8) .LBL 为标签文件, 它与 REPORT 有些类似, 只能是字符串变量, 其他类型需要转换。标签文件由 CREATE LABEL <文件名.LBL> 建立。

(9) .TXT 为正文文件, 用 COPY TO <文件名> SDF 或 SET ALTERNATE TO <文件名> 建立。亦可用 REPORT FORM 文件名.FRM TO FILE 文件名.TXT 建立。

§ 1.4 数据库的结构

数据库文件的结构是对库中每个字段的描述确定的。每个字段由字段名、字段类型和字段宽度来描述。

字段名 它最多可达 10 个字符 (即 5 个汉字), 要以字母为开头, 中间不能嵌入空格。字段名中允许出现字母、数字和下划线。

字段类型 dBASE III 允许有五种字段类型, 即字符型(C)、数字型(N)、逻辑型(L)、日期型(D)和备注 MEMO 型(M)五种, 每种类型简述如下:

字符型字段: 它可接收从键盘上打入的可打印的 ASCII 字符 (包括空格)。字段最大宽度是 254。

数字型字段: 该字段又分为整型和小数型两种, 前者是不含小数点的数。它的字段宽度为允许容纳的数字位数, 小数点作一位数字位。字段精度为 15.9 位 (不含小数点)。

逻辑型字段: 只能接收代表真 (TRUE) 或假 (FALSE) 的单一字符。真用 T、t、Y、y 表示, 假用 F、f、N、n 表示。

日期型字段: 用于存储日期, 输入的格式可定义成多种形式 (用 picture 描述), 缺省的格式为 mm/dd/yy (mm 表示月份, dd 表示日期, yy 表示年份), 其字段宽度为 8 个字符。日期可以参加运算, 即一个日期可以加减一个日期, 或加减一个数。

备注型字段: 为某些较大的文本信息而设计的, 它存储在数据库文件的辅助文件中, 它的大小是可变的, 如果键入数据, 字段位置每块为 512 字节, 最大可为 4096 字节。在每个数据库文件记录中, 每个 MEMO 字段用 10 个字节表示。

§ 1.5 关于内存变量

所谓内存变量, 就是在数据库结构外独立存储的数据项。它可以暂存数据, 亦可进行运算, 给程序编制带来方便。

内存变量有四种类型, 即数字型、字符型、日期型和逻辑型。dBASE III 允许有 256 个有效的内存变量, 最多可占用 6000 字节。内存变量名可以长达 10 个字符, 它必须以字母为开头, 由字母、数字和下划线组成。用户可任意定义内存变量。当一个内存变量名和现用数据库文件中的字段名相同时, 字段名优先于内存变量名。一般在同名变量前加字符 M- 来代表内存变量。程序中的内存变量可用 SAVE 命令把它们保存在磁盘文件中 (扩展名是 .MEM), 从磁盘文件中重新调入内存变量用 RESTORE 命令。

数字型内存变量以存储数字, 其精度可达 15.9 位。字符型内存变量以存储字符串, 最多可达 256 个字符。日期型内存变量以存储日期, 其长度为 8 个字节, 它们一般表示为

dd/mm/yy (欧洲格式), 这里 mm 表示月份, dd 表示日, yy 表示年。一个日期可以减去另一个日期, 结果是一个数字, 该数为两个日期之间的天数。一个数 (表示天数) 也可加上或减去一个日期, 其结果为一日期。逻辑型内存变量以存储真 (.T., .t., .Y., .y.) 或假 (.F., .f., .N., .n.) 值, 占一个字节。

在程序中出现的内存变量, 如果没有用全局变量 PUBLIC 命令描述, 当程序终止时, 将全部释放掉程序建立的内存变量。用 PUBLIC 标识的任何一个内存变量, 必须用 RELEASE 命令删除, 否则是删不掉的。内存变量不能重名。

§ 1.6 表达式的概念

所谓表达式, 就是由字段变量, 内存变量, 常数, 函数和操作运算符组成的式子称为表达式。dBASE III 能对表达式进行操作, 大大丰富了它的功能。

由操作运算符的不同, 表达式又可分为算术表达式, 逻辑表达式和字符表达式三种。

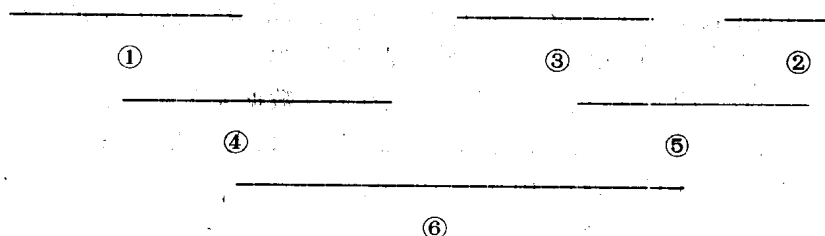
算术表达式是由数字型字段变量, 数字型内存变量, 数值常数, 数值函数以及算术运算符组成, 其值为一数值。

算术运算符有: + 加 - 减
* 乘 / 除
** 或 ^ 乘幂 () 圆括号, 用于改变运算次序

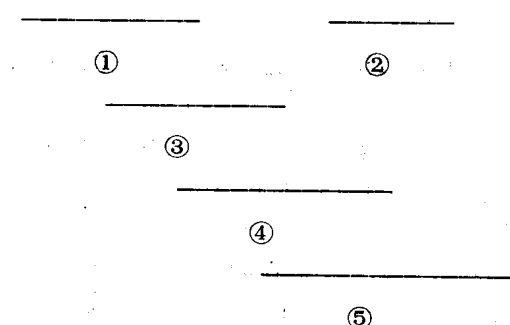
它们的优先级依次是: 单目的 + 和 -, 乘幂 (** 或 ^), 乘除 (* /), 加减(+ -), 先括号内, 后括号外, 同一级中从左到右执行。

例如:

$(35.4 + \text{SQRT}(27)) * \text{EXP}(A) - \text{LOG}(17.84) * * 2 / (16 + C)$



$(\text{数量} * \text{价格}) / 30 - A * * 3 + 17.25$



小圆中的数字表示运算次序。

其中: 数量, 价格为数字型字段变量, A、C 为数字型内存变量。

逻辑表达式是由算术表达式, 逻辑字段变量, 逻辑内存变量, 逻辑常数, 逻辑函数, 关

系运算符，逻辑运算符和字符串运算符组成。

关系运算符有：

<	小于	>	大于
=	等于	<>	不等于
>=	大于等于	<=	小于等于

它们无优先次序，从左到右执行。

逻辑运算符有：

.NOT. 逻辑非(对单一表达式起作用) .AND. 逻辑与
.OR. 逻辑或 () 圆括号，改变运算次序

它们的优先级依次是逻辑非(.NOT.)，逻辑与(.AND.)，逻辑或(.OR.)。若有括号，则先括号内，后括号外。

字符串运算符有：

+ 连接运算符，把两个或两个以上的字符串连接成一个字符串。

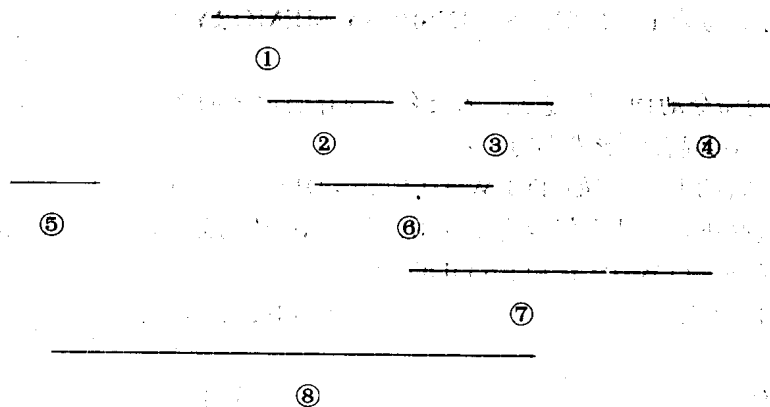
\$ 子串比较。(例如，如果 A，B 是字符串，若 A 和 B 相等或 B 包含 A，则 A\$B 的值为真，否则为假值)

它们无优先级次序，从左到右执行。

如果在同一表达式中使用了几种运算符时，则类型的优先级为：

- (1) 算术运算符和字符串运算符；
- (2) 关系运算符；
- (3) 逻辑运算符。

例如：A > B .OR. (257 - B) / C = A + B .AND. S1 \$ S2



小圆中的数字表示运算次序。

其中：A，B，C 为数字变量，S1 和 S2 为字符串变量。

§ 1.7 工作区的选择

dBASEIII 允许同时使用 1 到 10 个数据库文件，每个文件可独立地使用。在某一文件上工作时，任何其他文件上的当前记录可以读，但不能进行修改。

不同文件用不同的工作区来分隔。选择工作区用 SELECT 命令，并用数字 1 到 10 或字母 A 到 J 来标识 10 个工作区。工作区可任意选择。一进入 dBASEIII，就认为是工作区 1。

例如：.USE AB1

```

SELECT 2
USE AB2
SELECT 5
USE AB3

```

这时当前的工作区是 5, 现用文件 AB3 可读可修改。如执行:

```
SELECT A (或 SELECT 1)
```

此时, 现用文件是 AB1, 对 AB1 可读可修改。如执行:

```
SELECT 2 (或 SELECT B)
```

这时, 现用文件是 AB2, 对 AB2 可读可修改。

§ 1.8 某些规则

在使用 dBASE III 命令时, 要遵循下列几点规则:

(1) 一个命令的结构就是它的语法。每个命令以一个动词开始, 其后可跟随一个或多个子句。命令的一般格式如下:

```
动词 [<范围>][<表达式表>][FOR <条件>] WHILE <条件>]
```

方括号中的内容是可选项, 与出现的先后次序无关。实际命令中方括号和尖括号不需要键入。

(2) 命令的最大度为 254 个字符(包括各单词间的空格)。

(3) 命令中的单词和某些关键字可用前四个字符表示, 以便减少书写量和输入量。例如: INSERT BLANK 可简写为 INSE BLAN, DISPLAY MEMORY 可简写为 DISP MEMO 等等。

(4) dBASE III 中的命令动词、关键字、字段名、内存变量名和文件名等, 可以用英文字母的大写、小写或大小写混合的形式书写。

(5) 要注意语句间的配对, 如 DO WHILE...ENDDO, DO CASE...ENDCASE, IF...ENDIF, 所有这些语句彼此之间要全部配对。例如 IF 语句如果在 DO 语句的循环内, 则 IF 和 ENDIF 必须在 DO WHILE 和 ENDDO 之间, 不能交叉。

```
DO WHILE <条件>
```

```
:
```

```
IF <条件 1>
```

```
:
```

```
ENDIF
```

```
:
```

```
ENDDO
```

是正确的

```
DO WHILE <条件>
```

```
:
```

```
IF <条件 1>
```

```
:
```

```
ENDDO
```

```
:
```

```
ENDIF
```

是错误的

§ 1.9 符号的约定

在命令和语句的描述中, 常常用到一些符号, 其含义分别说明如下:

<...> 尖括号中的内容由用户提供;

[...] 方括号中的内容可任选;

/ 斜杠表示两个选择项中的任一个选择;

Active FILE 表示当前选择的工作区中的数据库文件;

Alias 数据库或字段的任选别名;

Character 表示一个字符, 可以是一个字符, 可以是一个字母或一个数字或一个符号;

String 字符串, 可以用引号或其他定界符括起来;

Column 表示显示或打印机上的起始列号;

Command 表示 dBASE III 的命令;

Condition 表示条件表达式, 其值为真、假;

Cstring 字符串的缩写;

Default 缺省设备或缺省值;

Delimiters 定界符, 字符数据一般是用双引号或单引号来定界;

Drive 表示驱动器的标识符;

Exp 表示表达式, 由内存变量、字段变量、函数、常数和运算符组成;

exp C 表示字符型表达式;

exp D 表示日期型表达式;

exp L 表示逻辑型表达式;

exp N 表示数值型表达式;

exp list 表示有二个以上的表达式, 表达式之间用逗号分开;

Field 数据库文件中的信息项, 亦称字段变量;

Field list 由二个以上的字段变量组成, 字段变量之间用逗号分开;

Field Name 字段名;

Field Type 表示字段变量的数据类型。字符型(C), 数字型(N), 日期型(N), 逻辑型(N)和备注型(M);

Field Width 字段宽度;

File 表示磁盘文件;

File Name 表示磁盘文件名;

FOR<Condition> 表示选择短语, 当条件成立时, 执行相应的功能;

Key 关键字;

Memvar 内存变量, 之间用逗号分开;

n 为指定的一个数值;

Parameter 表示参数;

Print 表示打印机;

Procedure 表示过程语句;

Row 表示显示器或打印机的行号;

Scope 指选择的范围: ALL 表示全部, RECORD n 指第 n 个记录, NEXT n 指从当前指针位置向下 n 个记录, 缺省值为当前记录。如有 FOR/WHILE 选择项时, 指 ALL;

Skeleton 泛指符, 又称通配符, 有? 和* 两种, ? 为单个字符的泛指符, * 为多个字符的泛指符;

To 表示后面是一个文件名或内存变量;