

Fox BASE⁺

程序设计实用教程

孙 蓉 梁柳青 编著



南开大学出版社

内 容 提 要

这是一本介绍 FoxBASE⁺ 的各种命令、文件及程序设计全过程的实用性很强的教科书。包括 FoxBASE⁺ 的主要性能、数据库文件操作、输入输出程序设计、查询修改程序设计、统计与汇总程序设计、排序与复制新的数据库程序设计、数据的更新连接程序设计、菜单程序设计、FoxBASE⁺ 与其他语言接口及程序设计应用实例等。

本书的最大特色是,将命令与程序设计融为一体,按照程序模块分别介绍。它以实例讲解,深入浅出,易于为读者较快掌握编程的基本方法。

书中各章末附有上机操作练习和检测题,以备读者检验学习效果。

本书可作为大专院校数据库课程的教材或参考书以及各类数据库培训班的教材或参考书。

FoxBASE⁺ 程序设计实用教程

孙蓉 梁柳青 编著

南开大学出版社出版

(天津八里台南开大学校内)

邮编 300071 电话 3508542

新华书店天津发行所发行

天津宝坻第四印刷厂印刷

1997 年 1 月 第 1 版	1998 年 1 月第 2 次印刷
开本: 787×1092 1/16	印张: 12.25
字数: 307 千	印数: 8001—10000

ISBN 7-310-00986-x
TP. 63 定价: 14.00 元

序

计算机技术的迅猛发展,使计算机应用成为跨越国界、进行国际交流、推动全球经济与社会发展的重要手段。随着社会的进步,愈来愈显示出了计算机与人们的经济生活、文化生活等诸多方面的密切关系。计算机知识已是我国当代知识分子知识结构中不可缺少的重要组成部分,乃至已成为全民的普遍要求。因此,我国除各级各类学校普遍设有计算机课外,各种计算机培训班也如雨后春笋般不断涌现,一个全方位、多层次地培养各行各业计算机应用人员的热潮已经形成。

关系型数据库管理系统 FoxBASE⁺自产生以来,就以其强大的功能深受好评。尤其是它容易使用的特点,更受到人们的青睐。目前, FoxBASE⁺已成为我国开发大、中、小型管理软件的主要工具之一。实践证明, FoxBASE⁺有着强大的生命力和广阔的应用前景。因此,各校计算机专业及其相关专业普遍将数据库列为必修或选修课。各种各样的计算机考试也都将它列为程序设计语种之一。

本书作者根据多年的数据库教学经验和长期的工作实践,对 FoxBASE⁺进行了全面、细致的探讨与研究,编写出了这本教程。以往大多数介绍 FoxBASE⁺的教材,都是用大量篇幅介绍命令的功能及语法格式,而后介绍 FoxBASE⁺程序设计。而本书改变了这一写法,其特点是以程序设计为核心,按程序模块分类,将介绍命令与介绍程序设计结合起来,便于读者理解、记忆各种命令,也便于读者掌握 FoxBASE⁺编程的方法。本书为了增强实践性,各章末附有上机实验,指明实验内容、步骤与方法以及要达到的考核目标。此外,它还附有极具针对性的习题与思考题。各章末还有小结,用简洁的语言归纳了一章内容的重点和难点。

本书可作为大专院校数据库课程的教材或参考书,也可作为各类数据库培训班及社会各界人士自学数据库的教材或参考书,还可供准备参加全国和各地方计算机等级考试的考生使用。

祝贺本书的出版,并希望本书对推动计算机教育与普及做出贡献。

陶龙芳

1996年12月于北京

前 言

随着微型计算机的广泛应用, FoxBASE⁺数据库管理系统已被越来越多的微机用户知晓, 并且以其强大优越的数据处理能力日益受到广大用户的欢迎。

通过多年的数据库课程的教学及编程设计实践, 我们感到编写一本集教学、程序设计、上机实验一体化的教科书很有必要。传统的有关数据库的教材对系统的各种命令、函数的功能等介绍得较为详细, 而对编程上机实践介绍得较少; 一些介绍 FoxBASE⁺的教材则多半是大量命令的堆砌令初学者望而却步。对于传统教材, 老师们感到介绍的内容大都是命令、语法规则等, 讲解具体命令时很少与程序设计有实际联系, 上机实验时也需老师逐条地讲解。学生通过这种方式学习, 无法掌握 FoxBASE⁺程序设计的基本方法, 往往是只停留在了解使用若干条命令的水平上, 离独立编制一些应用程序还存有相当大的距离。

鉴于上述原因, 我们参照有关资料并结合自己几年来的数据库教学与编程设计实践, 编写了这本《FoxBASE⁺程序设计实用教程》。本书从章节安排上作了精心设计, 使读者按照章节顺序学下去, 既能学到 FoxBASE⁺的各种命令、文件, 又能掌握程序设计的基本全过程。

本书的特点是将命令与程序设计融为一体, 按照程序模块分别进行介绍, 尽量把各程序模块中用到的命令集中讲解, 并且抓住一般常用命令格式中的短语进行介绍; 对于不太常用的其他命令短语, 读者可在本书学习的基础上通过查阅命令表进一步熟悉使用。本书的另一特点是每章都有较多的、步骤详细的上机操作指导练习, 并配有检测题, 便于读者在学习检验学习效果。本书旨在通过实例讲解各种命令的使用, 以增强实践性, 让读者学完本书后, 能尽快掌握编程的基本方法, 并最终能独立编写应用程序。

书中介绍以 FoxBASE⁺2.10 版为主。书中各章的程序举例和上机实验程序均采用开放式结构, 全部程序都在微机上调试通过, 读者可以此作为程序设计课题的参考。

全书共分十一章。第一章、第九章、第十章由梁柳青编写; 其余各章由孙蓉编写, 最后由孙蓉负责完成全书统稿。

在本书的编写过程中, 得到了何丕廉教授、任长明教授、刘锡海教授的支持与帮助, 他们对本书的编写提出了宝贵的意见, 并对本书作了较好的评价。天津广播电视大学有关任课教师也对本书提出了若干修改意见。在此, 一并向他们表示由衷的感谢!

本书可作为大专院校数据库课程的教材或参考书以及各类数据库培训班的教材或参考书。同时可作为广大数据库用户的辅助参考资料。

由于成书时间仓促, 作者水平有限, 书中错误难免, 敬请读者批评指正。

编者

一九九六年十月

目 录

第一章 概述	(1)
1.1 FoxBASE+要求的环境	(1)
1.2 FoxBASE+系统组成和主要性能指标	(1)
1.2.1 FoxBASE+系统组成	(1)
1.2.2 FoxBASE+主要性能指标	(2)
1.3 FoxBASE+的安装、启动和退出	(2)
1.3.1 安装	(2)
1.3.2 启动	(3)
1.3.3 退出	(3)
1.4 FoxBASE+的命令语句规则	(3)
1.5 FoxBASE+的文件	(4)
1.6 FoxBASE+的常量、变量及表达式	(6)
1.6.1 常量	(6)
1.6.2 变量	(6)
1.6.3 表达式和运算符	(7)
1.7 FoxBASE+的函数	(9)
1.8 上机实验	(10)
1.8.1 实验内容与要求	(10)
1.8.2 实验步骤与方法	(10)
习题与思考题	(14)
本章小结	(15)
第二章 数据库文件操作	(16)
2.1 数据库文件	(16)
2.2 建立数据库结构	(17)
2.3 上机实验	(18)
2.3.1 实验内容	(18)
2.3.2 实验步骤与方法	(18)
习题与思考题	(28)
本章小结	(29)
第三章 程序文件	(31)
3.1 程序文件的建立与执行	(31)
3.1.1 建立、修改程序文件	(31)
3.1.2 执行程序文件	(32)
3.2 程序的三种基本结构	(32)
3.2.1 顺序结构的程序	(32)
3.2.2 选择结构的程序	(32)

3.2.3 循环结构的程序	(36)
3.3 上机实验	(39)
3.3.1 实验内容	(39)
3.3.2 实验步骤与方法	(39)
习题与思考题	(44)
本章小结	(45)
第四章 输入输出程序设计	(47)
4.1 输入程序设计	(47)
4.1.1 输入程序中的@命令	(47)
4.1.2 屏幕格式文件	(48)
4.1.3 添加记录程序设计	(49)
4.1.4 插入记录程序设计	(50)
4.2 输出程序设计	(51)
4.2.1 利用@命令制表输出	(51)
4.2.2 利用?和??命令输出表格	(53)
4.3 利用报表格式文件输出数据	(54)
4.3.1 建立报表格式文件	(54)
4.3.2 报表输出	(56)
4.4 上机实验	(57)
4.4.1 实验内容	(57)
4.4.2 实验步骤与方法	(57)
习题与思考题	(65)
本章小结	(65)
第五章 查询修改程序设计	(67)
5.1 查询程序设计	(67)
5.1.1 查询程序中用到的命令	(67)
5.1.2 索引查询程序	(69)
5.2 修改程序设计	(71)
5.2.1 修改程序设计中的命令	(71)
5.2.2 删除程序设计中的命令	(73)
5.3 数组及其应用	(74)
5.3.1 数组的定义	(74)
5.3.2 数组的使用	(74)
5.3.3 数组和库文件的数据传递	(75)
5.3.4 数组在程序设计中的应用	(78)
5.4 上机实验	(79)
5.4.1 实验内容	(79)
5.4.2 实验步骤与方法	(80)
习题与思考题	(84)
本章小结	(84)
第六章 统计与汇总程序设计	(85)
6.1 统计与汇总程序中的命令	(85)

6.1.1 统计记录个数	(85)
6.1.2 求和,求平均值	(86)
6.1.3 汇总	(87)
6.2 上机实验	(88)
6.2.1 实验内容	(88)
6.2.2 实验步骤与方法	(88)
习题与思考题	(90)
本章小结	(90)
第七章 排序与复制新的数据库程序设计	(92)
7.1 数据排序	(92)
7.2 复制新的数据库	(93)
7.2.1 同时复制结构与记录	(93)
7.2.2 单独复制库结构	(93)
7.3 成批添加数据库文件的记录	(94)
7.4 上机实验	(95)
7.4.1 实验内容	(95)
7.4.2 实验步骤与方法	(95)
习题与思考题	(100)
本章小结	(100)
第八章 数据的更新、连接程序设计	(101)
8.1 工作区的使用	(101)
8.1.1 工作区的概念和选择	(101)
8.1.2 工作区的互访	(102)
8.2 两个数据库文件间的更新	(103)
8.3 数据库文件的连接	(105)
8.4 上机实验	(106)
8.4.1 实验内容	(106)
8.4.2 实验步骤与方法	(106)
习题与思考题	(111)
本章小结	(112)
第九章 FoxBASE+ 菜单程序设计	(113)
9.1 传统菜单的设计方法	(113)
9.2 光带菜单程序设计	(116)
9.2.1 光带菜单命令	(116)
9.2.2 光带菜单设计举例	(117)
9.3 上拉和下拉式菜单	(117)
9.3.1 上拉菜单的设计命令	(118)
9.3.2 下拉菜单的设计命令	(119)
9.4 上机实验	(122)
9.4.1 实验内容	(122)
9.4.2 实验步骤与方法	(122)

习题与思考题.....	(126)
本章小结.....	(126)
第十章 FoxBASE+与其他语言的接口	(127)
10.1 通过文本文件交换数据.....	(127)
10.1.1 库文件复制成文本文件	(127)
10.1.2 文本文件的数据添加到库文件	(129)
10.2 调用外部程序的命令 RUN/!	(130)
10.3 上机实验.....	(132)
10.3.1 实验内容	(132)
10.3.2 实验步骤与方法	(132)
习题与思考题.....	(135)
本章小结.....	(135)
第十一章 FoxBASE+程序设计应用实例	(136)
11.1 分析.....	(136)
11.2 总体结构设计.....	(136)
11.3 系统中的数据库设计.....	(136)
11.4 各模块程序设计.....	(138)
11.5 本系统的特点.....	(169)
附录一 FoxBASE+(2.10 版)命令一览表	(170)
附录二 FoxBASE+函数一览表	(180)

第一章 概述

FoxBASE⁺是美国 Fox Software 公司开发的微机数据库语言。1987 年 2 月推出 1.0 版, 1987 年 7 月又进行了修改, 推出 2.00 版, 目前的最高版本是 1988 年 7 月推出的 FoxBASE⁺ 2.10 版。与 80 年代国内外流行最广的 dBASE III 相比, 其功能和性能都有重大的改进, 其运行速度高于 dBASE III 6~8 倍。

FoxBASE⁺有单用户和多用户两类版本, 现在国内使用的中文版 FoxBASE⁺是根据西文的多用户版本汉化开发而成的, 它有 2.0 和 2.10 两种版本。

1.1 FoxBASE⁺要求的环境

FoxBASE⁺是用 C 语言开发的, 因而可以在多种操作系统及机器环境下运行。从 IBM—PC/XT、AT、286、386、486 等微型计算机, 到国产的长城、东海、浪潮等系列微型机; 以及它们的兼容机种, 都能够运行 FoxBASE⁺。

在微型机上运行 FoxBASE⁺至少需要两个磁盘驱动器, 如一个软盘和一个硬盘。在 IBM—PC/XT 及其兼容机上运行 FoxBASE⁺至少需要 360KB 内存。如果内存在 650KB 以上, FoxBASE⁺则尽可能多占用内存。这样, 工作更快, 性能更好, 因为 FoxBASE⁺的快速运行是靠大量的内存来实现的。

1.2 FoxBASE⁺系统组成和主要性能指标

1.2.1 FoxBASE⁺系统组成

FoxBASE⁺有单用户和多用户两种版本, 两种版本的主要组成文件如下:

文件说明	单用户版本	多用户版本	备注
执行程序	FOXPLUS.EXE	MFOXPLUS.EXE	FoxBASE ⁺
覆盖程序	FOXPLUS.OVL	MFOXPLUS.OVL	系统的主要成分
帮助程序	FOXPLUS.HLP	FOXPLUS.HLP	FoxBASE ⁺ 系统的辅助成分
过程组合程序	FOXIND.EXE	FOXIND.EXE	
准编译程序	FOXPCOMP.EXE	FOXPCOMP.EXE	

表中前两项为最主要的程序, 其中执行程序是经常使用的, 在 FoxBASE⁺启动后全部驻留内存。而覆盖程序在 FoxBASE⁺启动时仅有一部分被装入内存, 其余的要到需要时才临时调入, 而且调入新内容时, 原在内存中暂不使用的那部分程序被覆盖。有了这两个主要程序, 就可进入 FoxBASE⁺管理系统。

在辅助成分中,准编译程序用于编译FoxBASE+的程序文件。该程序只能产生一种中间代码文件,仍须由FoxBASE+解释执行,不能够独立运行。这种编译执行的特点是缩短了程序长度,提高了程序的装入速度和执行速度,有利于程序的保密。

过程组合程序用于将若干较小的、分散的程序文件合并为一个较大的过程文件,借以提高运行速度和方便管理。帮助程序则用于提供联机帮助信息,以帮助用户正确使用FoxBASE+。

用单用户版本编写的软件可以在多用户版本下运行,用多用户版本编写的软件也可以在单用户版本下运行,单用户版本与多用户版本百分之百兼容。

1.2.2 FoxBASE+主要性能指标

FoxBASE+各类主要性能指标所允许的最大参数值和默认参数值如下:

每个数据库文件最大记录数	1000 000 000
每个记录最多字符数	4000
每个记录最多字段数	128
每个字段最多字符数	254
数字计算中最多有效位数	16
字符串最多字符数	254
命令行最多字符数	254
报表头最多字符数	254
索引关键字最多字符数	100
内存变量默认数	256
最大内存变量数	3600
最大数组数	3600
每个数组中最多元素数	3600
最多打开文件数	48
支持单用户 FoxBASE+ 最低 DOS 版本	2.0
支持多用户 FoxBASE+ 最低 DOS 版本	3.1

1.3 FoxBASE+的安装、启动和退出

1.3.1 安装

FoxBASE+系统的五个主要文件通常全部存放在一张高密软盘上,由此可以通过软盘驱动器来启动FoxBASE+。但通常都先把FoxBASE+系统盘上的文件全部安装到硬盘上,以后直接由硬盘启动。

安装时首先在硬盘建立FoxBASE+的子目录,常以FOX为该子目录名。具体步骤如下:

① md\fox ✓ (建立以FOX为子目录名的子目录)

② cd\fox ✓ (进入FOX子目录)

③ copy a: *.* ✓ (将FoxBASE+系统盘插入a驱动器,并将上面的FoxBASE+系统文件复制到FOX子目录中。)

另外,还需在硬盘根目录上建立好 CONFIG.SYS 操作系统配置文件,在 FoxBASE+ 子目录上建立 CONFIG.FX,即 FoxBASE+ 的特性配置文件,这两个文件可用任意字处理或编辑程序建立。

1.3.2 启动

为方便以后的使用,可先将执行程序 MFOXPLUS.EXE 或 FOXPLUS.EXE 改名为 FOX.EXE。

在操作系统提示符后键入:

- ① cd\fox ✓ (进入 FOX 子目录)
- ② fox ✓ (启动 FoxBASE+)

启动后,屏幕上显示出 FoxBASE+ 的圆点提示符“·”,它是 FoxBASE+ 的工作状态标志,在圆点提示符后,用户可键入要执行的命令。

1.3.3 退出

若要结束 FoxBASE+ 的运行,则在圆点“·”提示符下键入

· quit ✓

即可退出 FoxBASE+ 系统并且关闭所有打开的文件,返回到操作系统提示符下。

1.4 FoxBASE+ 的命令语句规则

在 FoxBASE+ 语言中,对数据的操作都是由命令来完成的,命令相当于一般高级语言中的语句。FoxBASE+ 的命令有一定的规则,命令必须以动词开头,后跟若干个顺序无关的子句(或称短语),如下的两个语句是等价的:

DISPLAY NEXT 10 FOR 职称="讲师"

DISPLAY FOR 职称="讲师" NEXT 10

命令行最多为 254 个字符,命令和关键字可用前 4 个字母的缩写形式,字母的大小写形式不限。命令的一般格式为:

动词 [范围] [表达式] [FOR/WHILE 条件]

在命令格式中,方括号中是任选部分,尖括号中的部分是必选部分,当输入命令时不要输入方括号和尖括号。斜线号表示其前后两个项目中选择其中一个项目。若有省略号,则表示前面项目可继续重复多次选择,项目和项目之间用逗号隔开。

[范围]是大多数命令可选择的短语,说明此命令动词对数据库记录作用的数量范围。有下列四种范围:

RECORD n —— 某单个记录;

NEXT n —— 从当前记录开始以下 n 个记录;

ALL —— 全部记录;

REST —— 从当前记录到最后一个记录。

[FOR/WHILE 条件]短语命令只能对符合条件的记录进行操作。FOR 子句对数据库的全部记录进行选择操作,不管条件是真是假,整个数据库搜索一遍。WHILE 子句对数据库

记录进行操作时,从当前记录开始,只要遇到某一个记录(条件)为假,命令则停止操作。命令中可允许两种子句一起使用,并配合[范围]子句一起对记录进行选择,以 WHILE 子句优先。

1.5 FoxBASE+的文件

在 FoxBASE+ 中,所有的数据和程序都是以文件的形式存储在磁盘上的,按照不同的用途,FoxBASE+ 提供了 9 种类型的文件,所有文件都是按名访问的。文件名由不超过 8 位且不含空格的字符串构成,扩展名主要用来区分文件的类型,除非用户另有指定,在文件建立时系统将自动给它们加上默认的扩展名。程序文件或屏幕格式文件各有两种默认扩展名,一种用于没有编译过的文件,另一种用于经过编译程序编译后所产生的文件。

1.5.1 数据库文件(.DBF)

数据库文件是 FoxBASE+ 中最重要的文件,存放数据库的结构和数据,最多由 10 亿个记录组成,每个记录最多可含 128 个字段。字段是最小的数据单位,字段名最多由 10 个字符组成,并必须以字母开头,其间不含空格,但可含数字和下划线,下划线必须嵌在中间。

在数据库运行期间的所有操作中,当需要该字段数据时,就引用该字段名。实际上,数据库的字段名就是一个专用变量名。

字段是存放数据的,数据是有类型的,字段类型就是指明该字段存放的数据类型。字段数据类型共有 5 种,它们是:

- ①字符型(C)——由字母、汉字、数字和其他特殊符号组成,长度最多为 254 个字节。
- ②数值型(N)——由数字、小数点和正负号组成,小数点算一位长度。宽度为 19 个字节。
- ③逻辑型(L)——只能输入真(Y/y/T/t)或假(N/n/F/f),宽度固定为 1 个字节。
- ④日期型(D)——一般按美国日期格式 MM/DD/YY(月/日/年)形式表示,宽度固定为 8 个字节。
- ⑤记忆型(M)——当字段的字符超过 254 个时,字符型字段的宽度已不够用,此时可把字段定义为记忆型。在 .DBF 文件中此字段宽度固定为 10 个字节,这 10 个字节不是用来存储该记忆型字段的真正内容,而是用来存放一个指针。这个指针指向存放真正字段内容,且与数据库文件同名的记忆文件(.DBT)。它在数据库文件中用 Memo(Memory)一词表示。此字段可输入 4096 个字符,把内容都存于辅助文件(.DBT)中。

1.5.2 数据库文件备注文件(.DBT)

数据库备注文件是 .DBF 的辅助文件,用以存储 .DBF 文件中记忆字段的字符。当用户建立的 .DBF 文件中包含有记忆型字段时,系统便自动生成一个同名的 .DBT 文件。在含有记忆型字段的 .DBF 文件打开或关闭的同时,与其同名的 .DBT 文件也同时被自动打开或关闭。当复制 .DBF 文件时,必须复制与其有关的 .DBT 文件,否则 .DBF 文件就不能再打开了。

对用户来说,在建立 .DBF 文件时,当字符个数少于 254 个时,用字符型字段而不用记忆型字段,能提高效率。

对一个 .DBF 文件,可有多个记忆型字段,所有记忆型字段的内容都存储在同一个 .DBT 文件中。

1.5.3 内存变量文件(.MEM)

内存变量文件用来存储内存变量的内容。当内存变量的内容需要保存时,可用 SAVE 命令建立一个内存变量文件,并最多可容纳 3600 个内存变量。当需要某一内存变量的内容时,用 RESTORE 命令把内存变量文件读到内存中使用。

1.5.4 索引文件(.IDX)

在 .DBF 文件中,记录是按其物理顺序进行排列的,即指记录按输入的顺序排列,记录号是顺序的。但我们常常希望记录能按另外的某种规则排列。例如,按某个数值型字段的数值大小、按某个字符型字段的字符 ASCII 码顺序或者按某个日期型字段的日期顺序来排列。有时实际情况要求 .DBF 文件按不同关键字段进行排序,比如学生成绩库,要按每门课成绩或几门课成绩来进行排序。解决这一问题最好的办法,就是建立各种索引文件。

索引文件可由 INDEX 命令建立,由 REINDEX 重新索引。

1.5.5 报表格式文件(.FRM)

报表格式文件是将当前数据库文件(.DBF)的内容按报表格式文件里的格式信息输出。报表格式文件由 CREATE REPORT 命令建立,并由 REPORT 命令输出。

1.5.6 标签文件(.LBL)

用户常常需要根据数据库内容打印标签,比如信封或某商品的价格标签。可由 CREATE LABEL 命令建立标签文件,并由 LABEL 命令输出标签。

1.5.7 屏幕格式文件(.FMT)

利用格式化输入语句(“@”语句)对屏幕输入格式进行再设计。格式文件可用 MODIFY COMMAND 建立,再用 SET FORMAT TO(格式文件名)来打开它。这时用 APPEND、EDIT 等命令进行屏幕编辑,不再是系统给出的格式,而是由格式文件所设计的格式。

1.5.8 命令文件(.PRG)

命令文件是由一系列 FoxBASE+ 的命令和函数以及程序设计语句所组成的文件,是用户为完成某一任务而编制的程序,是 ASCII 码文件。由 MODIFY COMMAND 命令或文本编辑软件来建立,并用 DO 命令执行之。

1.5.9 文本输出文件(.TXT)

文本输出文件主要用作 FoxBASE+ 与其他应用软件之间的数据接口文件,它是 ASCII 码文件。可由特殊形式的 COPY 命令将 .DBF 文件复制成标准的文本数据文件,供其他软件使用这些数据,也可由 APPEND FROM 命令将其他标准数据文件的数据追加到 FoxBASE+ 的 .DBF 文件中。还可以用 SET ALTERNATE 命令记录 FoxBASE+ 的处理活动过程。

在以上 9 种文件中,最主要的是数据库文件和程序文件。本书将对数据库文件和程序文件作重点介绍。

1.6 FoxBASE⁺的常量、变量及表达式

像其他高级语言一样,数据、运算符、表达式及函数是构成 FoxBASE⁺的基本元素。本节将介绍 FoxBASE⁺的常量、变量及表达式,而函数的内容将在下一节介绍。

1.6.1 常量

常量是一恒定值,是在程序运行进程中不变化的数据,有以下几种类型。

1. 数字型常量

数字型常量可以是整数或实数,例如:3.14, -3, 5, 8, 19 等等。

2. 字符型常量

字符型常量就是用定界符括起来的字符数字串。有三种定界符:单引号' '、双引号" "和方括号[]。字符串中可含空格,例如:'ABC', "A BC",若字符串中含有单引号或双引号,只要最外层定界符成对就可以,例如:'ABC"book"D'或"AB'CD"是对的。若是数字,加上引号后,即成为字符串,无值的大小概念。例如:"256"是一个字符串,而不是数字 256。

3. 逻辑型常量

逻辑型常量只有两个值,一个是真值,一个是假值。真值书写形式为.T.,.t.,.Y.,.y. 4 种之一;假值书写形式为.F.,.f.,.N.,.n. 4 种之一。

4. 日期型常量

日期型常量总是由 8 个字符组成,其形式为 mm/dd/yy(美国)。例如:08/18/96 表示 1996 年 8 月 18 日。若将日期型常量赋值给变量时,可用 CTOD 函数,例如:CTOD("08/18/96")。

1.6.2 变量

变量是可以任意改变的数据项,FoxBASE⁺有两种变量,即数据库字段变量和内存变量。数据库字段变量即字段名,是在建立数据库结构时定义的,共有 5 种类型。而内存变量是独立存在的变量,使用时可临时定义变量名,其类型随着所赋值类型的改变而改变。内存变量可以是数值、字符、逻辑和日期型 4 种类型之一。

内存变量名由 1~10 个字符、数字、下划线组成,其中必须以字母或汉字开头,使用下划线时需嵌在中间。当内存变量与字段变量同名时,系统优先字段变量。为了防止混淆,用户在访问内存变量时,可加用前缀 M->,例如:NAME 是内存变量名,也是数据库字段名,当访问内存变量时,可写成 M->NAME。

FoxBAE⁺内存变量最大配置为 3600 个,系统默认数为 256 个,但可以通过配置文件 CONFIG.FX 设置使此值在 128 到 3600 范围内改变。除字符型变量外,其他变量不占字节数,在系统初始化时已分配了足够的空间。

FoxBASE⁺增加了数组的功能。数组也是一种内存变量,它由 DIMENSION 命令定义。FoxBASE⁺只有一维或二维数组,数组中成员叫数组元素。数组元素的值可以是 C,N,L,D 型的数据。关于数组的应用在使用时另作详细介绍。

1.6.3 表达式和运算符

1. 表达式

表达式是由常量、变量、函数和运算符组成的式子。例如： $2 * R * 3.14$ 、 $XM = \text{"王刚"}$ 、 $\text{数学} > 90$ 、 $\text{AND. 外语} \geq 90$ 等等。

一个字段变量，一个内存变量，一个常数以及一个函数都是一个表达式的特殊形式。在 FoxBASE+ 语言的许多命令子句中带有表达式。例如： $\text{DISPLAY } [\langle \text{范围} \rangle] [\text{FOR} \langle \text{表达式} \rangle]$ 。

2. 运算符

FoxBASE+ 提供了 4 种运算符：算术、关系、逻辑和字符串运算符，它们是表示运算类型的符号。

根据不同的运算符及表达式的结果的取值，表达式可分为算术表达式、关系表达式、逻辑表达式、字符串表达式。

(1) 算术运算符及算术表达式

① 算术运算符

()	括号
* * 或 ^	乘方
*	乘
/	除
+	加
-	减

② 算术表达式

例： $(10+2)/4$	运算结果为 3
$40-6$	运算结果为 34
$-6 * 25$	运算结果为 -150
$9 * * 2$	运算结果为 81

所有算术运算符运算的数据必须是数值型，其运算结果为算术型的值。

(2) 关系运算符及关系表达式

① 关系运算符

<	小于
<=	小于等于
=	等于
>=	大于等于
>	大于
<>	不等于(或#)
==	精确等于

② 关系表达式

例： $\text{"BASIC"} = \text{"Pascal"}$	运算结果为假
$12 < > 13$	运算结果为真
$15 * 2 + 7 < 6 / 2 - 50$	运算结果为假

关系运算可以进行算术值比较，字符串比较，也可以进行日期大小比较。关系运算符两边

的数据类型要相同。字符串是根据其相应的 ASCII 的值进行比较的。当关系运算符两边的表达式的结果符合关系运算符所表示的意义时为真,否则为假。

英文字母比较大小是按其 ASCII 码值的大小进行比较的。如"BCD">"ABC"为真,因为字母 B 的 ASCII 的值为 66,A 的 ASCII 码值为 65,所以"B">"A"。在字符串比较中,两个串的字母依次进行比较,因此字符串"BCD">"ABC"。

汉字字符比较大小,先用相应的汉语拼音替代汉字,然后再按上述英文字母比较大小的规则进行比较。

例:"中">"美"为真(.T.),因为"ZHONG">"MEI"。

(3)逻辑运算符及逻辑运算表达式

①逻辑运算符

() 括号

.NOT. 逻辑非——否定其结果,运算符右边关系表达式条件成立,其值就为假,否则就为真。

.AND. 逻辑与——运算符两边关系表达式条件都成立,其值就为真,否则就为假。

.OR. 逻辑或——运算符两边关系表达式中,只要一边条件成立,其值就为真,否则就为假。

②逻辑表达式

例:5+7>6 .AND. -2<0 运算结果为真(.T.)

.NOT. 4=5-1 运算结果为假(.F.)

3*5<>5 .OR. 3/2<0 运算结果为真(.T.)

逻辑运算符运算的表达式必定产生真或假的值,逻辑表达式的值亦为真或假。

(4)字符串运算符及字符串表达式

①字符串运算符

十 加号两边的两个字符串完全连接,合并成一个新的字符串。

- 减号表示两串相并操作。与加号操作不同之处在于,首先将减号左端字符串尾部空格去掉,再与右边字符串相加。

\$ \$符号表示两字符串比较操作,其含义是运算符左侧字符串是否包含在右侧字符串之中,若包含,则运算结果就为真,否则就为假。

②字符串运算表达式

例:"A"+"B" 运算结果为"AB"

"A"-"B" 运算结果为"AB"

"ANTIC" \$ "FRANTIC" 运算结果为真

"FLOAT" \$ "TITANIC" 运算结果为假

(5)运算符的优先级

每一种运算符都有一定的优先级,这种级别控制了运算的执行顺序。其优先级分别为:

① ()圆括号内运算,函数

② +, -(单项目运算符)

③ *,* 或 ^

④ *,/

- ⑤ +, - (加减运算符)
- ⑥ <, <=, =, >=, >, <> (或 #), ==, \$
- ⑦ .NOT.
- ⑧ .AND.
- ⑨ .OR.

序号小的优先级别高,同级运算符顺序是自左至右。

(6)操作示例

. ? 36 * 8 + (74 - 4) / 5	求算术表达式的值
302	结果
. STORE 12 TO A, B, C	把 12 赋给 A, B, C 内存变量
. ? A > B	A 大于 B 否
. F.	否(假)
. ? (A + 1 > B). OR. (A < C)	计算逻辑表达式的值
. T.	是(真)

1.7 FoxBASE⁺的函数

函数是 FoxBASE⁺语言的重要组成部分,它是把应用中最常用的运算过程编制成子程序存放在系统中。我们把由系统提供的函数称为标准函数,由用户自己定义的函数称为自定义函数。每种函数具有一定的操作功能,因此函数也有操作数和操作结果值,分别称为自变量和返回值。函数与运算符构成表达式,像常量和变量那样,它可作为表达式的一个组成部分。FoxBASE⁺语言为用户提供了上百种丰富的标准函数。按其功能划分可分为 5 类:

1. 数值运算函数

包括数值计算(例如取整函数 INT(),求平方根函数 SQRT(),取模函数 MOD()等),数值比较(例如比较大小函数 MAX(),MIN()等)及将非数值型数据转换成数值型数据函数(例如将字符串转换为数值的 VAL()函数)等共 11 种函数。

2. 日期和时间函数

这类函数可用来进行年月日、星期、时间的查询,也可以进行日期转换与比较。例如:YEAR()函数测定当前年份,DAY()函数测定当前日,CTOD()函数把日期格式的字符串转换为日期型数据等共 14 种函数。

3. 字符串操作函数

此类函数共有 18 种,包括字符串的转换、生成、测试以及字符串操作等函数。例如:LEN 函数可测定字符串的长度,STR 函数可将数值表达式转换为字符串,LEFT,RIGHT 函数可进行子串操作。

4. 状态判断函数

包括对文件状态、记录状态及字符串首字符状态判断等 13 种函数。例如:EOF()函数用来判断是否到数据库文件尾,FILE()函数用来检查文件是否存在等。

5. 状态参数测试函数

这类函数具有对系统文件、输入键、功能键、显示器、打印机、内外存储器、系统软件及状态