

计算机系列教材

FoxBASE+ 语言程序设计

习题解析与上机指导

本书编委会 编著

UNDER PC

河南科学技术出版社

2FB

序

为了满足计算机教学的需要,我们特组织编写了“计算机系列教材”。

为了编写本系列教材,我们专门组成了计算机教材编辑委员会。编辑委员会负责在全省有关学校的教师中,组成各门课程的编写组。编写组制订各门课程的教学大纲,确定章节内容,分工编写,由各主编、副主编统稿,最后由主审审定。参与各册具体编写工作的作者都是工作在微型计算机教学、实验、辅导第一线上的经验丰富的教师。在编写中,我们始终贯彻“理论与实际相结合、系统与应用相结合、软件与硬件相结合”的方针,使本套教材具有内容简练、叙述清晰、实例丰富的特色。

“计算机系列教材”将分批出版,计划出版的有:

- | | |
|-----------------------------|-----------------|
| 《FoxBASE + 语言程序设计》 | 《微机操作技术》 |
| 《FoxBASE + 语言程序设计习题解析与上机指导》 | 《PASCAL 语言程序设计》 |
| 《QBasic 语言程序设计》 | 《数据库原理》 |
| 《QBasic 语言程序设计习题解析与上机指导》 | 《C 语言程序设计》 |
| 《财会电算化》 | 《计算机维护与维修》 |
| 《计算机组成原理》 | 《计算机网络》 |
| 《操作系统原理》 | 《常用工具软件》 |
| 《单片机原理及应用》 | 《计算机英语》 |

根据微型计算机发展情况和教学需求,我们将增删系列书目,并修订已出版的教材。由于我们水平所限,谬误和不当之处在所难免,敬请有关专家和广大读者批评指正。

计算机教材编辑委员会

1998 年 1 月

目 录

第一章 数据库管理系统概述	(1)
第一部分 内容概述	(1)
第二部分 习题解答	(5)
第三部分 思考与练习	(7)
第二章 FoxBASE+ 的数据类型及其运算	(8)
第一部分 内容概述	(8)
第二部分 习题解答	(14)
第三部分 思考与练习	(19)
第三章 数据库的基本操作	(21)
第一部分 内容概述	(21)
第二部分 习题解答	(24)
第三部分 思考与练习	(28)
第四章 数据库记录的定位和编辑	(29)
第一部分 内容概述	(29)
第二部分 习题解答	(32)
第三部分 思考与练习	(35)
第五章 数据库记录的查询和统计	(37)
第一部分 内容概述	(37)
第二部分 习题解答	(40)
第三部分 思考与练习	(43)
第六章 顺序结构程序设计	(46)
第一部分 内容概述	(46)
第二部分 习题解答	(48)
第三部分 思考与练习	(56)
第七章 选择结构程序设计	(57)
第一部分 内容概述	(57)
第二部分 习题解答	(58)
第三部分 思考与练习	(67)
第八章 循环结构程序设计	(69)

第一部分	内容概述	(69)
第二部分	习题解答	(70)
第三部分	思考与练习	(79)
第九章	数组	(82)
第一部分	内容概述	(82)
第二部分	习题解答	(83)
第三部分	思考与练习	(88)
第十章	子程序	(89)
第一部分	内容概述	(89)
第二部分	习题解答	(92)
第三部分	思考与练习	(97)
第十一章	格式输出输入和菜单程序设计	(99)
第一部分	内容概述	(99)
第二部分	习题解答	(102)
第三部分	思考与练习	(108)
第十二章	多数据库文件的操作	(111)
第一部分	内容概述	(111)
第二部分	习题解答	(113)
第三部分	思考与练习	(118)
第十三章	数据库辅助操作命令	(122)
第一部分	内容概述	(122)
第二部分	习题解答	(124)
第三部分	思考与练习	(129)
第十四章	程序辅助控制命令	(130)
第一部分	内容概述	(130)
第二部分	习题解答	(132)
第三部分	思考与练习	(135)
第十五章	系统命令	(139)
第一部分	内容概述	(139)
第二部分	习题解答	(140)
第三部分	思考与练习	(142)
第十六章	综合练习	(143)
上机指导		(147)
上机指导一	FoxBASE+ 的启动和常用函数的使用	(147)
上机指导二	数据库文件结构的建立和修改、记录的输入和显示	(148)
上机指导三	数据库记录的定位和编辑	(149)
上机指导四	数据库记录的查询和统计	(150)
上机指导五	顺序结构程序设计	(151)

上机指导六	选择结构程序设计.....	(152)
上机指导七	循环结构程序设计.....	(153)
上机指导八	数组的设计.....	(154)
上机指导九	子程序、用户定义函数的设计和过程的调用	(154)
上机指导十	格式输入输出命令的使用和菜单技术.....	(155)
上机指导十一	多数据库文件的操作.....	(156)
上机指导十二	数据库的复制、记录的批量追加、库结构的间接建立.....	(159)
上机指导十三	程序辅助控制.....	(161)
上机指导十四	系统命令的使用.....	(162)
上机指导十五	程序的调试.....	(163)

第一章 数据库管理系统概述

第一部分 内容概述

一、数据库的基本知识

1. 数据处理

数据处理是指对数据进行收集、存储、分类、计算、加工、检索、传输和制表等处理的总称。数据处理的目的是为了对原始数据进行加工,从而得到我们所需要的有价值的数。利用计算机对数据进行处理经历了三个阶段:手工管理、文件管理和数据库管理。

2. 数据库的特点

数据库是以一定的组织方式存储在一起的、能为多个用户共享的、独立于应用程序的、相互关联的数据集合。数据库的数据具有共享性、独立性、完整性和冗余少的特点。

3. 数据模型

数据模型是在计算机中表示客观事物及其联系的数据及结构。根据组织方式的不同,目前常用的数据模型有层次模型、网状模型和关系模型三种。

关系模型中每个关系对应一张二维表,采用二维表来表示数据及其联系,表格与表格之间通过相同的栏目建立联系。

4. 数据库系统及其组成

数据库系统是指采用数据库技术后的整个计算机系统。它由以下内容组成:

- (1)计算机硬件系统:包括主机、键盘、显示器、硬盘驱动器、打印机等。
- (2)计算机软件系统:包括系统软件(操作系统、数据库管理系统等)和应用软件。
- (3)数据:存放在数据库的各个数据文件中,它保存在外存储设备(如磁盘、磁带)上。
- (4)用户:包括数据库管理员和终端用户。

5. 数据库管理系统的功能

(1)数据库的定义和建立:按照用户的要求定义数据库的结构,并在计算机内部真正建立一个加载数据的数据。

(2)数据库的操作:接受、分析、执行用户对数据库中数据的存取要求,通常包括插入、删除、更新和检索等操作。

(3)数据库的运行控制:控制数据库的运行和用户的并发性访问。

目前微机上比较流行的数据库管理系统有 FoxBASE +、FoxPro、dBASE V 等, 它们都是关系数据库管理系统。

6. 数据库应用系统

数据库应用系统是指数据库系统和各种方便操作的应用程序的总称。其中, 应用程序是指对数据库中的数据进行各种处理的程序, 它由程序员或用户编写, 用来调用数据库中所存储的数据。

二、关系数据库的基本概念

1. 关系、元组、属性、关键字

在数据库管理系统中, 通常把一个没有重复行和重复列的二维表格看成一个关系。表格中的每一行在关系中称为一个元组(对应数据库中的记录)。表格中的每一列在关系中称为一个属性, 每个属性都要有一个属性名(对应数据库中的字段名)。记录中的一个字段的取值, 称为字段值或分量。记录值随着每一行记录的不同而变化。

为了在关系中区分不同的元组, 若表中的某单个或属性组的值能唯一地确定一个元组, 则称该属性组为候选关键字。并且, 一个候选关键字必须包含使得它能够在整个关系中是唯一的所必需的最小属性数。若一个关系有多个候选关键字, 则选定其中一个作为主关键字。如果一个关键字只用一个单一的属性, 则称为单一关键字; 如果是用两个或多个属性, 则称为组合关键字。

在多个关系中, 若某属性或属性组不是当前关系的关键字, 但它是另一个关系的关键字, 则称这个属性或属性组是另一个关系的外部关键字。关系数据库中的各关系模式, 就是通过主关键字与外部关键字相互联系的。

表与表之间的关系实际就是按照某一关键字(主关键字和外部关键字)来确定一个表中记录与另一个表中记录之间的关系的。根据一个表中的记录与另一个表中的记录之间的对应数量关系, 分为一对一关系、一对多(或多对一)关系、多对多关系。

2. 关系模式

关系模式是对关系的描述。它包括关系名、组成该关系的各属性名、属性向域的映象、属性之间数据的依赖关系等。属性向域的映象常直接说明为属性的类型、长度。

3. 关系数据库

关系数据库是通过关系模式的集合及各关系之间的联系, 来描述数据之间的联系而建立的数据库。在关系数据库中, 一行对应一个“记录”, 一列对应一个“字段”, 每一个关系(表格)中的数据都存放在一个数据文件中, 若干个数据文件就组成了关系数据库。

4. 常用的三种关系操作

(1) 选择: 是指对数据库文件中某些指定的元组(记录)进行数据操作, 从中挑选出满足某些条件的若干元组(记录)组成新关系, 也称为“筛选”。

(2) 投影: 是指对数据库文件中某些指定的属性(字段)进行操作, 从中挑选出指定的若干属性(字段)组成新关系。

(3) 连接: 是指从两个关系中选取属性间满足一定条件的元组组成新关系, 也称为“联结”或“连结”。

5. 关系的性质

- (1)列是同质的,即每一列中的分量是同一类型的数据,来自同一个域。
- (2)不同的列可出自同一个域,每一列称为属性,要给予不同的属性名。
- (3)列的顺序无所谓,即列的次序可以任意交换。
- (4)任意两个元组不能完全相同。
- (5)行的顺序无所谓,即行的次序可以任意交换。
- (6)每一分量必须是不可分的数据项。

三、FoxBASE+ 的文件类型及其默认的文件扩展名

- (1)数据库文件: .DBF。
- (2)数据库备注文件: .DBT。
- (3)索引文件: .IDX。
- (4)内存文件: .MEM。
- (5)程序文件: .PRG。
- (6)格式文件: .FMT。
- (7)文本文件: .TXT。
- (8)标签文件: .LBL。
- (9)报告文件: .FRM。
- (10)程序文件(已编译的): .FOX。
- (11)格式文件(已编译的): .FMX。

其中,数据库文件、程序文件、内存文件和文本文件是独立的文件,而数据库备注文件、索引文件、格式文件、标签文件、报告文件则是由数据库派生出来的文件。

根据文件的操作特点,通常又将 FoxBASE+ 的文件分为结构式文件和非结构式文件。如命令文件(.PRG)、文本文件(.TXT)和格式文件(.FMT)属于非结构式文件,它们的文件内容可以用 DOS 的 TYPE 命令显示出来。而数据库文件(.DBF)、内存文件(.MEM)、索引文件(.IDX)、数据库备注文件(.DBT)、标签文件(.LBL)、报告文件(.FRM)、已编译的程序文件(.FOX)和格式文件(.FMX)属于结构式文件,这些文件的结构都有一定的框架,都要用专门命令才能调用。

四、FoxBASE+ 命令的结构

FoxBASE+ 命令通常由两部分组成。第一部分是命令动词,也称关键字,它的词意指明了该命令的功能。第二部分包含有几个跟随在关键字后面的短语,这些短语通常用来对所执行的命令进行某些限制性的说明。

在 FoxBASE+ 命令中,有许多是对数据库文件的记录进行操作的命令。其基本结构格式为:

命令动词 [[范围]][FIELDS <字段名表>][FOR/WHILE <条件>][OFF][TO PRINT]

1. <范围>子句

该子句表示本命令对数据库文件进行操作的记录范围,一般有四种选择:

- (1)ALL: 对数据库文件的全部记录进行操作。
- (2)NEXT <n>: 只对包括当前记录在内的以下 n 个记录进行操作。

- (3) RECORD $\langle n \rangle$: 只对第 n 个记录进行操作。
- (4) REST: 自当前记录开始到文件尾的所有记录。

2. FIELDS 子句

该子句规定当前处理的字段或表达式, 有以下三种格式:

(1) FIELDS $\langle$ 当前数据库字段名表 $\rangle$: $\langle$ 当前数据库字段名表 $\rangle$ 中只能包含当前数据库字段。

(2) FIELDS $\langle$ 字段名表 $\rangle$: $\langle$ 字段名表 $\rangle$ 中可以同时包含当前数据库和其他工作区中数据库的字段。

(3) FIELDS $\langle$ 表达式表 $\rangle$: $\langle$ 表达式表 $\rangle$ 中可以包括任何合法的表达式。

3. FOR/WHILE $\langle$ 条件 $\rangle$ 子句

FOR $\langle$ 条件 $\rangle$: 指在规定的 $\langle$ 范围 $\rangle$ 中, 按 $\langle$ 条件 $\rangle$ 检查全部记录。即从第一条记录开始, 满足 $\langle$ 条件 $\rangle$ 的记录就执行该命令, 不满足者就跳过去继续搜索, 直到最后一条记录。若省略 $\langle$ 范围 $\rangle$ 则默认为 ALL。

WHILE $\langle$ 条件 $\rangle$: 指在规定的 $\langle$ 范围 $\rangle$ 中, 只要 $\langle$ 条件 $\rangle$ 成立, 就对当前记录执行该命令, 并把记录指针指向下一个记录, 一旦遇到一个不满足 $\langle$ 条件 $\rangle$ 的记录, 就停止搜索, 并结束该命令的执行。即遇到第一个不满足条件的记录时, 就停止执行该命令, 即使后面还有满足条件的记录也不再执行。若省略 $\langle$ 范围 $\rangle$ 则默认为 REST。

4. TO PRINT 子句

该子句可以附加在 DIR、DISPLAY、LABEL、LIST、REPORT 等命令中, 在显示输出的同时送到由 SET PRINTER TO 命令指定的或者系统默认的打印机、输出设备或文本文件。

5. OFF 子句

在 DISPLAY、LIST 命令中选择该子句时, 系统将不显示记录号, 省略时则显示记录号。

五、FoxBASE+ 的基本上机操作

1. FoxBASE+ 的启动

开机后, 首先用 DIR 命令查看是否已建立有 FOX 子目录(若没有, 则建立), 进入该子目录, 查看是否有下列文件 FOXPLUS.EXE、FOXPLUS.OVL 等, 若没有则拷贝上去。然后, 输入 FOXPLUS 执行命令。即

```
C:\ > CD FOX
```

```
C:\FOX > FOXPLUS
```

这时即进入 FoxBASE+ 系统。

2. 退出 FoxBASE+

格式: QUIT

功能: 结束 FoxBASE+ 的运行, 关闭所有文件, 返回操作系统。

3. 程序文件的建立

格式: MODIFY COMMAND [$\langle$ 命令文件名 $\rangle$]

功能: 生成和编辑程序命令文件, 可用于任何标准的 ASCII 文本文件。

说明:

(1)文件扩展名默认为 .PRG。

(2)[〈文本文件名〉]中除非指定驱动器符及路径,否则,只在当前磁盘目录中进行存取。

(3)以·W 或·End 存盘退出。如果用·Q 或 Esc 键退出,则编辑不生效,原来的文件也不改为备份文件。

4. 程序的执行

程序的执行方法有两种:

(1)在 DOS 状态下,执行 FOXPLUS [〈程序文件名〉]。

(2)在圆点状态下,使用命令 DO 〈程序文件名〉。

第二部分 习题解答

1. 一个完整的数据库系统应有几部分组成?

答:数据库系统是指采用数据库技术后的整个计算机系统。一个完整的数据库系统应有下面四个部分组成:

- (1)计算机硬件系统:包括主机、键盘、显示器、软硬盘驱动器、打印机等。
- (2)计算机软件系统:包括系统软件(操作系统、数据库管理系统等)和应用软件。
- (3)数据:存放在数据库的各个数据文件中,它保存在外存储设备(如磁盘、磁带)上。
- (4)用户:包括数据库管理员和终端用户。

2. 一个关系模型应满足什么条件?

答:一个关系模型应满足下列条件:

- (1)每一列(字段)中的分量是同一类型的数据,来自同一个域。
- (2)不同的列可出自同一个域,每一列的属性名不能相同。
- (3)列的顺序无所谓,即列的次序可以任意交换。
- (4)任意两个元组不能完全相同。
- (5)行的顺序无所谓,即行的次序可以任意交换。
- (6)每一分量必须是不可分的数据项,即每一字段必须是不可再分的基本字段。

3. FoxBASE+ 是一种 A 数据库,其库文件的基本操作是由 B、C、D 三种基本运算组合而成的。B 是从关系表中抽出特定的记录;C 是从关系表中抽出特定的字段;D 是将二张以上的关系表相应连成新表。

供选择的答案:

- | | | | | |
|------|------|------|------|------|
| A: | ①层次型 | ②网状型 | ③关系型 | ④数值型 |
| B~D: | ①统计 | ②投影 | ③选择 | ④索引 |
| | ⑤拷贝 | ⑥连接 | ⑦排序 | |

答:FoxBASE+ 是一种关系型数据库管理系统,其库文件的基本操作由投影、选择、连接三种。

选择是指对数据库文件中某些指定的记录进行数据操作,从中挑选出满足某些条件

的若干记录组成新关系。

投影是指对数据库文件中某些指定的字段进行操作,从中挑选出指定的若干字段组成新关系。

连接是指从两个关系中选取字段间满足一定条件的记录组成新关系。

4. FoxBASE+ 中规定的文件类型有几种? 其中数据库文件、索引文件、备注文件和程序文件的扩展名是什么?

答: FoxBASE+ 中规定的文件类型有 11 种, 它们是: 数据库文件、数据库备注文件、索引文件、内存文件、程序文件、格式文件、文本文件、标签文件、报告文件、程序文件(已编译的)和格式文件(已编译的)。

其中, 数据库文件的默认扩展名为 .DBF, 索引文件的默认扩展名为 .IDX, 备注文件的默认扩展名为 .DBT, 程序文件的默认扩展名为 .PRG。

5. FoxBASE+ 规定的文件类型中, 哪些文件可用 DOS 的 TYPE 命令查看文件的内容?

答: 根据文件的操作特点, FoxBASE+ 的文件分为结构式文件和非结构式文件。非结构式文件有命令文件(.PRG)、文本文件(.TXT)和格式文件(.FMT), 它们的文件内容可以用 DOS 的 TYPE 命令显示出来。

6. FoxBASE+ 的命令行中的哪些成分可以简写? 最少写几个字符? 命令中的哪些成分的顺序是任意的?

答: 在 FoxBASE+ 的命令行中, 命令动词和子句中的短语可以用其前 4 个以上字符缩写表示, 最少要写前 4 个字符。例如: DISPLAY STRUCTURE 可简写为 DISP STRU。

命令行中, 大多数命令动词后跟一个或多个限定该动词的子句。命令行中的各个子句其顺序是任意的, 可以按任意次序排列。

7. FoxBASE+ 命令中的命令动词、关键字、文件名、变量名用大写字母和小写字母书写, 是否等价? 命令动词和各短语之间以空格分隔, 其中空格数目有限制吗?

答: FoxBASE+ 命令行的内容可以用英文字母的大写、小写或大小混写, 也就是说, FoxBASE+ 命令中的命令动词、关键字、文件名、变量名用大写、小写或大小混写都可以, 都是等价的。命令行中各个词应以一个或多个空格隔开, 命令动词和各短语之间的空格数目没有限制。如果两个词之间已嵌有双引号、单引号、括号、逗号等分界符, 则空格可以省略。但应注意, .T. 或 .F. 两个逻辑值中的小圆点与字母之间不许有空格。

8. FoxBASE+ 是一种关系型数据库管理系统, 命令中的哪些短语可用于关系的选择操作? 哪些短语可用于关系的投影操作?

答: 在关系操作中, 选择是指对数据库文件中某些指定的记录进行数据操作, 从中挑选出满足某些条件的若干记录组成新关系。而投影是指对数据库文件中某些指定的字段进行操作, 从中挑选出指定的若干字段组成新关系。

在 FoxBASE+ 命令中, 有许多是对数据库文件的记录进行操作的命令。其基本结构格式为:

命令动词 [**<范围>**][**FIELDS <字段名表>**][**FOR/WHILE <条件>**][**OFF**][**TO PRINT**]

其中,〈范围〉短语表示本命令对数据库文件进行操作的记录范围, FIELDS 〈字段名表〉短语规定当前处理的字段或表达式, 在 FOR/WHILE 〈条件〉短语规定的〈范围〉中, 按〈条件〉检查记录。

因此, 〈范围〉短语和[FOR/WHILE 〈条件〉]短语是用于关系的选择操作。[FIELDS 〈字段名表〉]短语是用于关系的投影操作。

第三部分 思考与练习

1. 数据库有什么特点?
2. 目前常用的数据模型有哪几种?
3. 简述关系、记录、字段、关键字的含义。
4. 为什么把 FoxBASE+ 数据库管理系统称为关系型数据库管理系统?
5. FoxBASE+ 命令的书写规则主要有哪些?
6. 简述 FoxBASE+ 的上机操作过程。

第二章 FoxBASE + 的数据类型及其运算

第一部分 内容概述

一、数据和数据类型

数据是信息的表达方式, 计算机中的数据都属于一定的数据类型, FoxBASE + 系统定义了六种类型的数据, 即数值型数据、字符型数据、逻辑型数据、日期型数据、备注型数据、屏幕型数据。

1. 数值型数据

数值型数据是指可以进行数学运算的数据。数值型数据只能包含 0~9 数字、小数点、正负号。数值型数据用 N 来表示。

2. 字符型数据

字符型数据是指一切可打印的字符和字符串。它包括 26 个英文字母、10 个数字、各种符号和空格、汉字以及由这些符号组成的字符串等。字符型常量要用定界符(双引号、单引号或方括号)括起来。字符型数据用 C 表示。

3. 逻辑型数据

逻辑型数据是用来作为各种逻辑判断的条件。逻辑型数据只有两个值: 真(True)和假(False), 有时又称: 是(Yes)和非(No)。逻辑型数据用 L 表示。

4. 日期型数据

日期型数据用来表示日期。日期通常采用美国格式 mm/dd/yy(月/日/年)。日期型数据用 D 表示。

5. 备注型数据

备注型数据是一种长度可变的字符型数据, 是为了存储备注信息而设计的。它们存储在数据库的辅助文件 .DBT 中。在数据库文件结构中, 备注型数据用 M 表示。

6. 屏幕型数据

屏幕型数据只用于保存屏幕显示信息。屏幕型数据用 S 表示。

二、常量和变量

1. 常量

常量是指在整个程序运行过程中, 始终保持不变的数据。FoxBASE + 定义了四种类

型的常量:数值型常量、字符型常量、逻辑型常量、日期型常量。

常量的数据类型由其书写格式自动确定。数值型常量是由正负号、数字和小数点组成的小数或整数。字符型常量是由双引号或单引号或方括号括起来的一串字符,也称字符串。逻辑型常量是用小数点夹住的字母表示,如 .T.、.t.、.Y.、.y.、.F.、.f.、.N.、.n.。日期型常量必须通过函数 CTOD() 进行转换,一般要写成 CTOD("月/日/年年")的形式。

2. 变量

变量是指在程序执行的整个过程中,其值可以变化的数据。

变量名是指变量在内存中所占的存储单元的名字。变量名是由用户自己定义的,通过相应的内存变量名来使用内存变量。FoxBASE+ 规定变量名最多为 10 个字符,可以由字母、汉字、数字和下划线组成,但必须以字母或汉字开头,中间不能有空格。

变量的当前值是指当前时刻在变量中存放的数据的值,简称变量值。

(1)内存变量:内存变量是指在程序中存放原始数据、中间结果或最终结果的临时性变量。FoxBASE+ 中内存变量的建立和赋值是同时完成的。

FoxBASE+ 规定内存变量的类型有五种:数值型、字符型、逻辑型、日期型和屏幕型。前四种类型的变量可以进行相应的某些运算;而屏幕型内存变量是专用于保存和恢复屏幕的,不能进行其他运算。

(2)字段变量:字段变量只能依附于数据库文件,也称数据项、字段名。FoxBASE+ 的字段变量类型有五种:数值型字段、字符型字段、逻辑型字段、日期型字段、备注型字段。

每个字段变量的数据类型在建立数据库时,由专门的命令 CREATE 进行定义和说明。

字段变量的值等于数据库文件的当前记录中对应字段的值。如果当前记录改变了,则所有和该数据库文件相应的字段变量的值将全部自动换成新的当前记录对应字段的值。

三、表达式

FoxBASE+ 的表达式是指符合 FoxBASE+ 规定,用运算符和括号将常量、内存变量、字段变量或函数连接起来的式子。

根据表达式值的数据类型,把表达式分为数值型、字符型、关系型、逻辑型、日期型表达式。

1. 数值型表达式

数值型表达式由数值型运算符、数值型常量、内存变量、字段变量和函数组成,其结果为数值型数据。数值型运算符有 +、-、*、/、和()。各个运算符进行运算的优先顺序为先乘除后加减,括号优先,同级运算符,按自左向右的规则进行运算。

表达式的书写规则如下:

(1)每个符号占一格,所有符号都必须一个一个地并排写在同一横线上,不能在右上角或右下角写方次或下标。

(2)原来在数学表达式中省略的内容必须重新写上。

(3)所有括号都用小括号(),括号必须配对。

(4)要把数学表达式中的有些符号,改成 FoxBASE+ 可以表示的符号。

2. 字符型表达式

字符型表达式由字符型运算符、字符型常量、内存变量、字段变量和函数组成。其一般格式为:

〈字符串 1〉〈字符串运算符〉〈字符串 2〉

字符型运算符只能用于字符型数据,对字符型数据进行运算操作。字符型运算的结果仍然是字符型数据。字符型运算符有字符串连接符(+)、串尾空格移位连接符(-)两种。

字符型运算符只有一种优先级,自左向右进行运算。

3. 关系型表达式

关系型表达式由关系型运算符、数值型表达式、字符型表达式组成。但关系运算符两侧的数据类型必须完全一致。关系型表达式的格式为:

〈表达式 1〉〈关系运算符〉〈表达式 2〉

关系运算符用于对两个相同数据类型表达式值的大小进行比较。被比较的数据可以是数值型、字符型、日期型。其运算结果为逻辑型数据真.T.或假.F.。

关系运算符有: <、<=、=、>、>=、<>(或#)、\$ 和 =。关系运算符均单独使用,不存在优先级问题。

数值型数据按其数值大小进行比较;日期型数据将日期看成“yymmdd”的六位整数,按数值大小比较;字符型数据,若是单个字符,按其 ASCII 码值大小比较;汉字字符按区位码大小比较。

4. 逻辑型表达式

逻辑型表达式由关系型表达式、逻辑型运算符、逻辑型常量、内存变量、字段变量和函数组成。其一般格式为:

〈关系表达式 1〉〈逻辑运算符〉〈关系表达式 2〉

逻辑型运算符用来对逻辑型数据进行各种逻辑操作运算。逻辑型表达式运算的结果仍然为逻辑型数据,即真.T.或假.F.。逻辑运算符有 .NOT.(或!)、.AND. 和 .OR.。

逻辑运算符的优先级为:先运算括号内的数据,其次为 .NOT.、.AND.、.OR.。逻辑运算符两边的圆点不可省略。

5. 日期型表达式

日期型表达式由算术运算符(+或-)、算术表达式、日期型常量、内存变量、字段变量和函数组成。日期型数据是一种特殊的数值型数据,它们之间只能进行加(+)、减(-)运算。日期型数据有以下三种运算:

(1)求两个日期相差的天数:两个日期型数据可以相减,结果是一个数值型。

(2)向后推算日期:把一个表示天数的数值型数据加到日期型数据中,其结果仍然为一日型数据。

(3)向前推算日期:把一个表示天数的数值型数据从日期型数据中减掉它,其结果仍然为一日型数据。

四、常用内部函数

1. 数学运算函数

(1) 取整数 INT()

格式: INT(<数值表达式>)

功能: 截去数值表达式值的小数部分, 函数值为整数部分。|数值型|

(2) 取两数中较大数 MAX()

格式: MAX(<数值表达式 1>, <数值表达式 2>)

MAX(<日期表达式 1>, <日期表达式 2>)

功能: 求出指定的两个数值表达式或两个日期表达式中较大的一个。|数值型/日期型|

(3) 取两数中较小数 MIN()

格式: MIN(<数值表达式 1>, <数值表达式 2>)

MIN(<日期表达式 1>, <日期表达式 2>)

功能: 求出指定的两个数值表达式或两个日期表达式中较小的一个。|数值型/日期型|

(4) 取两数整除的余数 MOD()

格式: MOD(<数值表达式 1>, <数值表达式 2>)

功能: 求出两个数相除得到的余数, 该函数又称求模函数。|数值型|

(5) 四舍五入 ROUND()

格式: ROUND(<数值表达式 1>, 数值表达式 2)

功能: 对指定的数值进行四舍五入。|数值型|

(6) 求平方根 SQRT()

格式: SQRT(<数值表达式>)

功能: 求出数值表达式的平方根。|数值型|

2. 字符操作函数

(1) 宏代换 &

格式: &<内存变量>[.<字符表达式>]

功能: 用于代换一个字符型内存变量的内容。

说明: 用于宏代换的字符型内存变量必须是标量变量, 而不能是数组或数组中的元素。&<内存变量>与跟在它后面的其他字符之间, 一般要隔一个或多个空格。如果代换后的值需要与后面其他字符连在一起, 则必须在 &<内存变量>之后与其他字符之间插入一个“.”。

(2) 在字符串中检索子字符串 AT()

格式: AT(<字符表达式 1>, <字符表达式 2>)

功能: 给出第一个字符串在第二个字符串中的起始位置。|数值型|

说明: 如果第二个字符串中不包括第一个字符串, 则函数值为 0。

(3) 从字符串左端取子字符串 LEFT()

格式: LEFT(<字符表达式>, <数值表达式>)

功能:给出指定字符串中最左面给定个数的字符。|字符型|

说明:字符个数由<数值表达式>确定,如果大于原字符串长度,则函数值为整个字符串,如果小于或等于0,则函数值为一个空串。

(4)从字符串右端取子串 RIGHT()

格式:RIGHT(<字符表达式>,<数值表达式>)

功能:给出指定字符串最右面给定个数的字符。|字符型|

说明:字符个数由<数值表达式>确定,如果大于原字符串长度,则函数值为整个字符串,如果小于或等于0,则函数值为一个空串。

(5)求字符串长度 LEN()

格式:LEN(<字符表达式>)

功能:求指定字符串的长度。|数值型|

(6)将大写字母转换为小写字母 LOWER()

格式:LOWER(<字符表达式>)

功能:把指定字符串中的所有大写字母转换成小写字母。|字符型|

(7)生成空格串 SPACE()

格式:SPACE(<数值表达式>)

功能:根据指定的空格数产生一个空格字符串。|字符型|

(8)在字符串中取子串 SUBSTR()

格式:SUBSTR(<字符表达式>,<数值表达式1>[,<数值表达式2>])

功能:求指定字符串中的子串。|字符型|

说明:子串在字符串中的起始位置由<数值表达式1>决定,长度由<数值表达式2>决定,省略<数值表达式2>时,长度从起始位置一直到字符串的最后一个字符。如果<数值表达式1>等于0,或者<数值表达式2>等于0,本函数值均为一个空串。<数值表达式1>的取值范围是从0至字符串的长度。

3. 类型转换函数

(1)将字符串转换为日期 CTOD()

格式:CTOD(<字符表达式>)

功能:将指定的字符串转换为日期型。|日期型|

说明:字符串的默认格式是:mm/dd/yy,可以用SET DATE和SET CENTURY ON/OFF命令来改变这一默认格式。如果字符表达式书写了不正确的格式或日期数字不合理,将产生一个空的日期型数据。

(2)将日期转换成字符串 DTOC()

格式:DTOC(<日期表达式>[,<1>])

功能:将日期表达式转换为字符串。|字符型|

说明:日期显示的格式由当前SET DATE的选择决定。如果用选择项<1>,则函数用yyymmdd格式,以适应在索引中直接使用,这对于按日期顺序维护数据库特别有用。

(3)将数值转换为字符串 STR()

格式:STR(<数值表达式1>[,<数值表达式2>[,<数值表达式3>]])