

FOXBASE+

● 陈世波 编著

实用编程



人民邮电出版社



计算机技术丛书

FoxBASE⁺实用编程

陈世波 编著

人民邮电出版社

内 容 提 要

本书主要介绍多用户关系数据库 FoxBASE⁺ 的编程方法和技巧。全书共分 16 章,分章讲述了 FoxBASE⁺ 的基本概念、函数、语句、命令、库文件、库结构、索引操作、数据统计、编程方法、程序调试、程序结构化设计、万能条件查询程序设计、多用户的程序设计、程序的编译,在最后一章给出了一个实际的财务管理系统程序。书后附有 FoxBASE⁺ 函数集、FoxBASE⁺ 命令集和 EPSON 系列打印机控制命令。

本书内容实例众多、深入浅出、注重实际、适合大、中专院校师生和广大计算机爱好者阅读。

计算机技术丛书

FoxBASE⁺ 实用编程

◆ 编 著 陈世波

◆ 人民邮电出版社出版发行 北京崇文区夕照寺街 14 号
北京顺义兴华印刷厂印刷
新华书店总店北京发行所经销

◆ 开本: 787×1092 1/16

印张:18

字数:445 千字

1997 年 8 月第 1 版

印数:1—8 000 册

1997 年 8 月北京第 1 次印刷

ISBN7-115-06434-2/TP·415

定价:24.00 元

丛 书 前 言

世界上发达国家普遍重视发展以计算机和通信为核心的信息技术、信息产业和信息技术的应用,一些经济发达国家信息产业发展迅速。

当前,我国处于国民经济高速发展时期。与此相伴随,必将有信息技术、信息产业和信息技术应用的高速发展。各行各业将面临信息技术应用研究与发展的大课题以及信息化技术改造的大任务、大工程。

为了适应信息技术应用大众化的趋势,提高应用水平,我们组织编写、出版了这套“计算机技术丛书”。这套丛书以实用化、系列化、大众化为特点,介绍实用计算机技术。

这套丛书采取开放式选题框架,即选题面向我国不断发展着的计算机技术应用的实际需要和国际上的实用新技术,选题不断增添又保持前后有序。

这套丛书中有的著作还拟配合出版软件版本,用软盘形式向读者提供著作中介绍的软件,以使读者方便地使用软件。

我们希望广大读者为这套丛书的出版多提意见和建议。

前 言

FoxBASE⁺是已为人们所熟知的多用户关系型数据库,众多的优点使它受到了计算机用户的广泛欢迎。对于初学 FoxBASE⁺的朋友来说,如何在编程时使用各类函数、命令、循环语句、嵌套包含的条件语句,是非常复杂和困难的。因此,学习语言本身仅仅是第一步,更重要的是掌握编程方法。如果只是一味地死记这些命令,却没有清楚的编程思路和熟练的编程技巧,也是无法顺利编出一个完整的 FoxBASE⁺程序的。

正因为上述原因,加之多年使用 FoxBASE⁺编程略有心得,我萌发了写《FoxBASE⁺实用编程》一书的想法。在这本书中不是单纯地讲解各种命令和语句,而是偏重于实用技巧,在全书的讲解中共结合了 100 多个实用程序,这些程序全都在 FoxBASE⁺下运行通过。通过阅读这些程序,读者可以细心领会程序设计的奥妙方法和精细技巧。

全书共分 16 章。第一至第三章介绍了 FoxBASE⁺的基本概念、数据、命令和语句等;第四章详细讨论了索引文件及快速查找命令,为数据库系统编程入门打下初步的基础;第五章讲述了数据的统计;第六章介绍命令文件和开关命令;第七章介绍格式化输出/输入命令;第八章介绍程序控制语句;第九章详细地讲解了程序的设计方法,还介绍了数据结构的重点:排序与检索;第十章介绍程序的调试;第十一章讲述程序的结构化设计;第十二章是一个完整的 FoxBASE⁺程序设计范例;第十三章给出了万能条件查询的程序设计;第十四章将大家从单用户设计带入到了多用户设计的世界;第十五章介绍如何把一个源程序编译成目标程序;第十六章是一个有实用价值的财务管理系统程序,全面而细致地介绍了程序设计的思路和步骤。

在编写本书的过程中,得到了许多同志的帮助和支持,特别是蔡庆波同志对本书提出了许多宝贵意见,在此深表感谢!

由于作者水平有限,加之时间仓促,书中错误之处在所难免,敬请广大读者批评指正。

作 者

目 录

第一章 数据库基本概念	1
1.1 简介	1
1.2 FoxBASE+基础	2
1.2.1 文件组成	2
1.2.2 启动	3
1.2.3 文件及文件名	4
1.2.4 工作盘设定	4
1.2.5 数据库结构	4
1.3 数据库文件的建立	7
1.3.1 清屏命令	7
1.3.2 建立库结构	7
1.3.3 显示磁盘文件	9
1.3.4 数据库文件的打开及关闭	9
第二章 FoxBASE+数据	11
2.1 常数	11
2.2 变量	11
2.2.1 字段变量	12
2.2.2 内存变量	12
2.3 表达式	14
2.3.1 数值型表达式	15
2.3.2 字符型表达式	16
2.3.3 逻辑型表达式	16
2.4 函数	18
2.4.1 数值型函数	18
2.4.2 字符型函数	22
2.4.3 逻辑型函数	24
2.4.4 日期型函数	25
第三章 库文件、库结构及记录操作命令	28
3.1 指针定位命令	28
3.1.1 指向命令	28
3.1.2 相对移位命令	28
3.1.3 定位命令	29
3.1.4 继续定位命令	29
3.2 显示记录命令	30
3.2.1 连续显示记录命令	30

3.2.2 分页显示记录命令	32
3.3 增加记录命令	32
3.3.1 追加记录	32
3.3.2 插入记录	33
3.4 修改记录	33
3.4.1 EDIT 编辑命令	34
3.4.2 BROWSE 窗口编辑命令	34
3.4.3 CHANGE 修改命令	34
3.4.4 REPLACE 替换命令	34
3.5 删除记录	35
3.5.1 给记录作删除标记命令	35
3.5.2 恢复删除命令	36
3.5.3 清除命令	36
3.6 文件操作命令	37
3.6.1 显示文件	37
3.6.2 文件复制	37
3.6.3 文件改名	38
3.6.4 删除文件	38
3.7 库结构操作命令	38
3.7.1 建立库结构	39
3.7.2 修改库结构	39
3.7.3 显示或打印库结构	39
第四章 数据库索引操作	42
4.1 索引库文件	42
4.1.1 索引文件的建立	43
4.1.2 索引文件的打开	45
4.1.3 索引文件的关闭	47
4.2 库记录的查询	47
4.2.1 快速查询命令	47
4.2.2 查询命令比较	49
第五章 数据的统计	51
5.1 计算记录数	51
5.2 数值型字段求和	51
5.3 计算数值型字段平均值	52
第六章 命令文件及其它	53
6.1 工作区选择	53
6.2 命令文件	54
6.2.1 命令文件的建立和修改	54
6.2.2 命令文件的调用	55
6.2.3 命令文件显示及打印	56

6.3 开关命令	56
6.3.1 SET TALK ON/off	56
6.3.2 SET PRINT on/OFF	56
6.3.3 SET DEVICE TO print/SCREEN	56
6.3.4 SET STATUS ON/off	56
6.3.5 SET SCORBOARD ON/off	57
6.3.6 SET CONSOLE ON/off	57
第七章 格式化输出/输入命令	58
7.1 格式清屏命令	58
7.2 格式画边框命令	58
7.3 格式化输出/输入命令	59
7.3.1 在指定位置上显示表达式	59
7.3.2 在指定位置进行格式输入	63
7.3.3 菜单显示格式	69
第八章 程序控制语句	73
8.1 程序流程图、框图符号	73
8.2 条件语句	74
8.3 多分支选择语句	79
8.4 循环语句	81
第九章 程序设计方法	94
9.1 显示素数的程序设计	94
9.2 找出规律来设计程序	97
9.3 排序方法	103
9.3.1 选择排序	103
9.3.2 冒泡排序	105
9.3.3 线性插入排序	106
9.3.4 对半插入排序	107
9.4 检索	109
9.4.1 线性检索	109
9.4.2 对半检索	110
9.5 数据库文件的查找	113
9.6 保密口令设定	117
9.6.1 最常见方法:建立一个库文件 KLK.DBF	117
9.6.2 对 KLK.DBF 库中口令进行变换	118
9.6.3 利用日期函数 DATE()进行口令设置	121
9.7 屏幕格式编程方法	122
9.7.1 让用户输入表格中的内容	122
9.7.2 带阴影的条式菜单格式显示	123
9.7.3 镶嵌菜单格式显示	124
9.7.4 利用清框画一条加框图案	125

9.7.5 利用 ASCII 码的字符画边框	125
9.7.6 利用 ASCII 码一个汉字显示出两种颜色	128
9.8 清屏程序设计	129
9.8.1 清颜色框图例	129
9.8.2 从上往下清整屏	129
9.8.3 从下往上清整屏	130
9.8.4 从左往右清整屏	130
9.8.5 从右往左清整屏	131
9.8.6 上、下同时往中间清整屏	131
9.8.7 左、右同时往中间清整屏	132
9.8.8 行从中间往上、下清整屏	132
9.8.9 列从中间往左、右清整屏	132
9.8.10 列从 39 列、79 列同时往左清整屏	133
9.8.11 列从 19 列、39 列、59 列、79 列同时往左清整屏	133
9.8.12 国际象棋图案清屏	134
9.9 UC DOS 汉字系统特殊显示	136
9.9.1 特显文字命令表	137
9.9.2 特殊显示命令表	138
第十章 程序调试	145
10.1 命令或语句编写、输入时的错误	145
10.2 变量没有赋初值	146
10.3 漏了 ENDIF、ENDCASE、ENDDO 等语句结束符	146
10.4 编写思路上的错误导致结果不对	147
10.5 调试程序方法	149
10.5.1 插入 WAIT 语句,显示中间变量值	149
10.5.2 插入调试程序命令	149
第十一章 程序的结构化设计	152
11.1 程序的结构化设计简介	152
11.2 子程序的建立与修改	153
11.2.1 子程序建立	153
11.2.2 子程序的结束语句及作用	153
11.2.3 子程序调用	153
11.2.4 带参数的子程序调用	154
11.3 过程文件	156
11.3.1 过程文件建立	156
11.3.2 过程文件打开	156
11.3.3 过程文件关闭	156
第十二章 数据库文件设计例题	162
12.1 总控程序 ZK.PRG	163
12.2 增加人员模块程序 ZJ.PRG	164

12.3	修改记录模块程序 SG. PRG	167
12.4	查询资料模块程序 CX. PRG	170
12.5	打印资料模块程序 DY. PRG	173
12.6	删除某人记录模块程序 SC. PRG	177
12.7	告初学者	179
第十三章	万能条件查询程序设计	183
13.1	用户选择条件设计	183
13.2	用户输入条件设计	193
第十四章	多用户的程序设计	195
14.1	加锁方法	195
14.1.1	数据文件的独占加锁	195
14.1.2	通过加锁函数对数据文件进行“写”加锁	195
14.1.3	操作命令自动建立“写”加锁	197
14.2	死锁的产生	198
14.3	死锁的预防	199
14.3.1	事先规定一个加锁顺序	199
14.3.2	一次对全部数据加锁	200
14.4	错误识别函数及命令	201
14.5	网络程序设计举例	201
14.5.1	三个库文件	201
14.5.2	程序设计模块	202
14.5.3	程序文件清单	203
第十五章	FoxBASE⁺、FoxPro 和 dBASE III 编译	226
15.1	FoxBASE ⁺ 伪编译	226
15.1.1	伪编译	226
15.1.2	反编译	226
15.2	FoxPro 2.5 编译系统	227
15.3	dBASE III 编译	229
第十六章	一个实际的财务管理系统应用	230
16.1	分析问题	230
16.2	确定操作步骤	236
16.3	根据操作步骤编写源程序	239
16.4	小记	260
附表一:	FoxBASE⁺函数集	261
附表二:	FoxBASE⁺命令集	264
附表三:	EPSON 系列打印机控制命令	273

第一章 数据库基本概念

1.1 简介

数据库是组织、存储、管理数据的电子仓库。在关系型数据库中,数据不是杂乱无章地存放,而是有一定规律的。我们可以把关系型数据库看作一个二维表格,该表格由行、列组成。以下是某单位的人事库:

编号	姓名	性别	出生年月	职称	工资	部门	婚否	工作日期
001	王伟鹏	男	07/12/66	工程师	598.36	一车间	.T.	09/10/88
002	刘虹艳	女	11/03/46	工人	762.00	二车间	.T.	05/25/65
003	彭建平	男	01/08/76	出纳	469.23	财务科	.F.	01/15/95
004	蔡清萍	女	02/30/59	工人	639.00	一车间	.T.	09/25/80
005	陈业同	男	12/09/43	厂长	937.20	厂部	.T.	01/23/66
006	王华	女	09/22/71	技术员	518.21	一车间	.F.	09/17/93

从上表可知,最上一行为表格项,共有九项,每项指明一个内容,在数据库中被定义为“字段名”。从第二行开始,每一行为一个人的档案,数据库称为“记录”。每一列上均反映相同类型的内容,比如,绝不会在“姓名”栏中,既有姓名,又有职称;而在“职称”栏中,既有职称,又有工资。为什么要用关系型数据库呢?因为关系型数据库简单、灵活、直观,很容易用形象的二维表格表达出来。另外从这张表格中还能看出数据库是由两部分组成:数据库结构、记录。拿一幢房子作比喻吧,现在的房子都采用框架结构设计,这幢“房子”如果看成一个“数据库”,那么房子的“框架”就是“数据库结构”,而“每套房间”就是一个“记录”。当我们建这幢房子时,先建好房子水泥柱的框架结构,这个结构不是建房子的目的,我们的目的就是要建好一套套房间,但这个“框架”的大小设计是根据每套房子的要求来确定的。我们再回头看上面的表格,这个表格也是有“框架”的,它有九个字段名,即编号、姓名等,如“编号”,它里面存放的内容就是每个人的编号,编号由3位数组成,编号由001、002、…、999组成,即可以表示1000人以下的单位。但如果该单位超过1千人、又小于1万人时,则3位数表示不下,就要用4位数来表示,当然4位上也允许。再看一下,表格中数据的“类型”也是有区别的,编号、工资都是由数字构成,但是工资是可以进行算术运算的(即工资可以加升、扣减),如800元,再涨300元工资,工资就为1100元了。但是编号不能进行算术运算,001号加上2,它能等于003号吗?

建立数据库时,只有建好了“库结构”,给出了字段名、数据类型和宽度大小(空间)后,才能往里输入记录,就像建好了仓库,再往里放原材料、成品一样。

有了“数据库”这个概念后,我们要学习一些什么内容呢?这就要紧紧围绕数据库、数据库结构、记录来学习,如显示数据库、库结构、记录的命令,增加库结构、记录的命令,修改库结构、

记录的命令,打印库结构、记录的命令,删除记录的命令,查询某种类型记录的命令等。

数据库管理系统是全面地实现数据库管理的一种软件,它的作用是维护数据库,接受和完成用户程序或命令提出的访问数据库的各种请求。用户的最终目的是使用和管理数据库,而数据库管理系统是帮助用户实现这一目的的工具和手段。

目前的数据库管理系统比较多,即使某种系统,也有多种版本。后开发的系统兼容先前的系统,即不改动或稍加修改就可在另一个系统上使用。其中 dBASE III 是早期广泛流行的数据库管理系统,它主要用在十六位微型计算机上,dBASE 为 Data BASE 的缩写,英文意思为“数据库”。dBASE IV 和 dBASE V 是在 dBASE III 基础上发展起来的。

dBASE III 为解释语言,美国 Nantucket 公司于 1985 年推出了编译 dBASE 版本,它将 dBASE 编写的程序编译成扩展名为 .EXE 的可执行文件,可以脱离 dBASE 而直接在 DOS 下运行,大大加快了程序的运行速度,同时对源程序起加密作用,防止非法修改和拷贝。在编译过程中,会对源程序作语法和逻辑上的检查,这样便于发现源程序的错误。

FoxBASE⁺是美国 Fox Software 公司开发的数据库管理系统,它完全兼容 dBASE 管理系统,运行速度高出 dBASE 好多倍,函数多了,可以定义数组,也能进行伪编译,它有单用户版和多用户版。

美国 Fox Software 公司 1991 年 7 月推出了 FoxPro 2.0 版,该产品以强大的功能获得广大用户的青睐。它面向对象和事件,其扩展版使用全部现有的扩展内存,是一个真正的 32 位的产品。它有 FoxPro 2.0 与新的 FoxPro 2.5 系列产品。FoxPro 2.0 系列包括交互式或开发式的单用户版、网络版及多用户版,每种版本又包含标准版和增强版。单用户版与多用户版之间的差异随着 FoxPro 2.5 的出现而消失,DOS 下的 FoxPro 2.5 和 Windows 下的 FoxPro 2.5 都是真正的多用户产品,它们既可在网络环境下运行,也可在单机下运行。

虽然数据库管理系统很多,但它们中的基本函数和语句是共有的,而且高版本兼容低版本,Foxpro、FoxBASE⁺、dBASE 等管理系统软件也是往下兼容的。以下我们将以 FoxBASE⁺ 的 2.00 版为基础来详细讲述数据库管理系统。

1.2 FoxBASE⁺ 基础

1.2.1 文件组成

多用户版本下的 FoxBASE⁺ 有如下 5 个文件:

MFOXPLUS.EXE	执行文件
MFOXPLUS.OVL	覆盖文件
FOXPHELP.HLP	帮助文件
FOXBIND.EXE	过程文件编辑器
FOXPCOMP.EXE	伪编译程序

单用户版本下的 FoxBASE⁺ 有如下 5 个文件:

FOXPLUS.EXE	执行文件
FOXPLUS.OVL	覆盖文件
FOXPLUS.HLP	帮助文件

FOXBIND. EXE 过程文件编辑器

FOXPCOMP. EXE 伪编译程序

注: dBASE III 只含两个文件: DBASE. EXE 与 DBASE. OVL

1.2.2 启动

在硬盘上建一个子目录存放 FoxBASE+ 文件, 方法如下:

C>MD FOX ; 建立 FOX 子目录

C>CD FOX ; 进入该子目录

C>COPY A:. ; 将 A 盘上装有 FoxBASE+ 的文件拷入 FOX 子目录下

另外根目录下系统配置文件 CONFIG. SYS 应有如下两条命令, 否则打不开过多的数据库文件。

FILES=30 ; 可同时打开 30 个文件

BUFFERS=10 ; 缓冲区为 10

启动步骤:

1. 先启动汉字系统

最好使用 UC DOS 希望汉字系统, 因为它不占用常规内存, 将更多的常规内存留给 FOXBASE+ 系统使用, 当然启动 WPS 金山汉字系统或 213 汉字系统也可以。

C>TYPE UC DOS. BAT ; 启动希望汉字系统

CD C:\UCDOS

RD16

KNL

PY

WB

RDSL

TX

C>TYPE WPS. BAT ; 启动金山汉字系统

PATH C:\WPS

SPLIBW

SPDOS

WBX

PRT24

2. 启动 FoxBASE+ 系统

C>CD FOX

C>MFOXPLUS

. ; 圆点(.)状态表示 FoxBASE+ 启动成功, 可以使用 FoxBASE+ 命令。

. QUIT ; 表示退出 FoxBASE+, 返回到 DOS 状态下。

1.2.3 文件及文件名

同 DOS 下的文件一样, FoxBASE⁺ 有 9 种指定的格式保存信息, 每种格式存放一定类型的文件。

文件名为 1~8 个字符, 是以英文字母或汉字开头, 由英文字母、汉字、数字及各种符号(中间不能嵌入空格)组成。

如: 下列文件名是正确的:

HT1 学生库 仓库_1 RSK GZ1 XS-K JS#5 HZ#K

下列文件名则是不正确的:

1AB _仓库1 人事管理库 CH EN RS? K XS* K

文件的类型符(扩展名)由 3 个字符表示, 文件名和类型符间用一圆点“.”分隔开, 一般情况下类型符是自动建立的, 下面列出这 9 种文件对应的类型符:

文件种类	类型标记(扩展名)
数据库文件	.DBF
数据库记忆文件	.DBT
索引文件	.IDX (dBASE III 下为 .NDX, 编译 dBASE 为 .NTX)
内存变量文件	.MEM
命令文件	.PRG
屏幕格式文件	.FMT
标签文件	.LBL
报表格式文件	.FRM
文本文件	.TXT

1.2.4 工作盘设定

设 FoxBASE⁺ 系统在 C 盘\Fox 子目录下, 如果数据库文件想存放在软盘或硬盘的其它地方, 就要用如下命令设置:

.SET DEFAULT TO <设备名>

这样以后数据文件读写就在指定的设备名上, 如果不设定, 仍在 FoxBASE⁺ 系统目录下读写数据。如设数据存放在 A 盘上:

.SET DEFAULT TO A: 或

.SET DEFA TO A

1.2.5 数据库结构

要建数据库文件, 必须先建数据库结构, 如同建造房子时, 先得建好房子的框架。只有建好了库结构, 才能往结构里输入记录。结构的建立又是严格按记录的要求来设计的。数据库结构包含四个部分: 字段名、类型、宽度及小数位。

1. 字段名

表示输入的数据项。如上表人事库中编号、姓名、性别等为字段名。

字段名定义: 为 1~10 字符, 一个汉字算两个字符, 它是以英文字母或汉字开头, 由英文字母、汉字、数字及下划线组成, 中间不能嵌入空格。如“姓名”字段, 你输“姓 名”是不正确的。字

段名的命名,最好同所反映的内容相一致,以便理解和记忆。比如,字段“性别”,它就表示是男的或是女的。为了减少对汉字的输入,特别是移植到 FoxPro 编译版上,它是不接受汉字字段名的,我们一般采用取汉字拼音首字母的办法。比如“姓名”取“XM”(即 Xing Ming 各首字母),这样也很谐音。注意应取半角下的英文字母,“XM”或“xm”都认为不妥,大小写字母都认为是一样的。下列汉字字段名对应简码字段名为:

编号	姓名	性别	出生年月	职称	工资	部门	婚否	工作日期
BH	XM	XB	CSNY	ZC	GZ	BM	HF	GZRQ

dBASE Ⅲ 允许一个库的最多字段数为 128 个,FoxBASE⁺则可达 4000 个。但实践证明,字段个数在 20 个左右最佳,如果字段太多,就分成几个库。

2. 类型

表示输入数据的数据类型,FoxBASE⁺有下列五种数据类型。

字符型类型:其内容是字符串,可以是英文字母、汉字、数字、特殊符号或空格。上表中编号、姓名、性别、职称、部门为字符型类型。字符型常数是用单引号括起来的,如'001'、'业彤'、'蔡煜'、'昔日重来'、'卡萨布兰卡'。

数值型类型:可进行算术运算的数据,其内容只能包含数字 0 到 9、小数点及正负号。上表中工资为数值型类型。

为什么编号不能定义成数值型类型?虽然编号也是由 0 到 9 组成,但编号不能进行算术运算,不能认为 001 号加上 007 号等于 008 号。另外 001 号同 1 号是不相同的,如果定义了数值型,001 号就是 1 了。而工资为什么又不能定义成字符型呢?其一:工资本身是一个算数;其二:工资是可以变化的,即可以调工资,可进行增减运算。数值型类型是很多的,如产品的数量、单价、库存量、销售金额等。

日期型类型:它用于存放表示日期的数据,形式为 mm/dd/yy,即以月/日/年的形式存放,如 1995 年 7 月 18 号,日期型表示为 07/18/95,月、日、年均以二位数表示,不足前面以 0 补充。上表中出生年月、工作日期为日期型类型。

日期型数据可进行如下运算:一个日期加上或减去一个十进制数得一个新日期型数据;两个日期数相减得一个相差天数;另可以进行日期比较。

上述日期形式为美国格式,按如下命令可以定义日期形式:

.SET DATE AMERICAN	美 国(默认格式)	MM/DD/YY
.SET DATE ANSI	美国国家标准协会	YY.MM.DD
.SET DATE BRITISH	英 国	DD/MM/YY
.SET DATE ITALIAN	意大利	DD-MM-YY
.SET DATE FRENCH	法 国	DD/MM/YY
.SET DATE GERMAN	德 国	DD.MM.YY

若不选择,则为美国格式。

✓有时也可把日期型定义成字符型,主要是因为索引库中多个关键字索引时,一定都要转化成字符型。

逻辑型类型:可用作各种逻辑判断,对只有“是”或“否”两种可能性的数据可以定义为逻辑型。如婚否、付款否、过期否等字段可定义为逻辑型。逻辑型只能有两个值,即逻辑真,用 .T.、.t. 或 .Y.、.y. 表示;逻辑假用 .F.、.f. 或 .N.、.n. 表示。

备注型类型:也称后备型类型,从本质上来说,它也是一种字符型,但字符型的最大宽度为

254 个字符,如果超过该宽度,只能定义成备注型类型,它最大可以到 4096 个字符,即 4K 字节。如工作简历、合同摘要、产品介绍,可以定义成备注型数据。如果在数据结构中定义了备注型,则有一个备注型文件,文件名同数据库文件,扩展名为 .DBT。如数据库定义为 CH.DBF,则数据库备注文件为 CH.DBT。

√如果编写程序来控制数据库,一般不定义成备注型,如果宽度超过 254 个字符,可以定义成二或三个字符型。侧如简历,可以定义简历 1、简历 2 等。

上述五种数据类型的类型标记如下:

字符型	Character	用 C 表示
数值型	Numeric	用 N 表示
日期型	Date	用 D 表示
逻辑型	Logical	用 L 表示
备注型	Memo	用 M 表示

3. 宽度

表示输入数据的数据宽度,下面列出五种数据类型的宽度范围:

字符型:	宽度 $\leq$ 254 字符	最大可定义到 254 个字符
数值型:	宽度 $\leq$ 19 位	该宽度由整数位+小数点(占 1 位)+小数位
日期型:	宽度=8	其宽度自动给出 8 位,即月月/日日/年年
逻辑型:	宽度=1	也是自动给出 1 位,它只有两个值,逻辑真与假
备注型:	宽度=10	在数据库文件中(.DBF)为 10 位,在备注文件中(.DBT)可达 4096 个字符。

解释:上述宽度只有字符型和数值型是由你确定的,确定的方法应从该类型内容中的最大长度去考虑。比如“姓名”字段,一般是 3 个汉字,即宽度为 6 个字符。但有的姓氏为“司马、欧阳、上官”等,其“姓名”字段就需要 4 个汉字,即宽度应给出 8 个字符。又比如“编号”字段,如果不超过 1000 人,则宽度为 3 位就够了。如果该单位已接近 1000 人,考虑到以后会增加人,这样宽度应给出 4 位。如果不增加宽度,用字母 A~Z 也是可以的。在设计库结构时,应有一定的弹性,因为库结构不仅是现在用,还要考虑今后。程序设计完全是针对库结构来设计的,如果程序设计完毕后,再返回来修改库结构中字段宽度,或增、减字段,这样改动的代价将是巨大的。

4. 小数位

只有数值型字段才有小数位。比如“工资”,小数位占 2 位,如果工资超过 1000 元,则宽度=整数位(4)+小数点(1)+小数位(2)=7 位。如果小数位为 0,则宽度就是整数部分的位数。

上述人事库建的库结构如下表:

字 段 名	类 型	宽 度	小 数 位
编号	C	3	
姓名	C	6	
性别	C	2	
出生年月	D	8	
职称	C	6	

续表

字 段 名	类 型	宽 度	小 数 位
工资	N	6	2
部门	C	6	
婚否	L	1	
工作日期	D	8	

1.3 数据库文件的建立

从现在开始学习 FoxBASE⁺ 命令, FoxBASE⁺ 命令是在圆点(.)状态下使用的, FoxBASE⁺ 命令有下列约定:

- (1) 必须以命令动词开头, 大小写字母都认为相同, 且只输入前四个字母就可以。
- (2) 带< >表示必选项, 带[]项是可选项, /格式表示两者中选择一个, ... 表示后面重复使用的格式。
- √(3) 命令字母、数值型数据、运算符等只能用半角字符。
- √(4) 命令一行写不下, 在该行尾写上“;”, 就可以往下一行写。
- (5) NOTE 或星号(*)开头的为注释, 它不执行, 只是方便阅读。
- (6) 每行程序后面加(&&.), 也表示注释, 它只适用 FoxBASE⁺ 系统下编写的程序。

1.3.1 清屏命令

格式: CLEAR

作用: 清除屏幕上的全部内容, 光标回到屏幕左上角。它类似 DOS 环境下的 CLS 命令。

如: 清除屏幕上的内容

. CLEAR

. CLEA

. clear

. cLEa

上述几种格式都是正确的, 允许只键入前边四个字母, 字母大、小写都是一样的。

1.3.2 建立库结构

格式: CREATE <库文件名>

作用: 建立一个新的数据库文件结构。库文件名不带扩展名时, 会自动加扩展名. DBF。你也可以加其它类型的扩展名, 如:

. CREA CHEN. DDD ; 这也是允许的, 但不合乎常理

现在建立人事库结构, 库结构名为 RSK. DBF

. CREATE RSK

FoxBASE⁺ 是英文提示的, 在进入结构输入时, 会出现光标、插入、删除、存盘等用法, 按