

计算机语言实用程序 与编程技巧

FOXBASE⁺
数据库语言

陈福盈 编 著
杨路明

王云宜 审 定

中南工业大学出版社



计算机语言实用程序与编程技巧

FOXBASE⁺ 数据库语言

陈福盈	编著
杨路明	
王云宜	审定

中南工业大学出版社

[湘] 新登字 010 号

计算机语言实用程序与编程技巧

FOXBASE+数据库语言

陈福盈 杨路明 编著

王云宜 审定

责任编辑：肖梓高

*

中南工业大学出版社出版发行

湖南大学印刷厂印装

湖南省新华书店经销

*

开本：787×1092 1/16 印张：12 字数：302千字

1994年11月第1版 1994年11月第1次印刷

印数：0001—7000

*

ISBN 7-81020-690-7/TP·041

定价：9.00元

本书如有印装质量问题，请直接与生产厂家联系解决。

《计算机语言实用程序与编程技巧》丛书

编委会成员

编委会主 任: 王云宜

编委会副主任: 田荣璋

编委会成 员 (以姓氏笔画为序):

王云宜	毛先成	田荣璋	向南平
任朝阳	陈福盈	杨路明	肖梓高
吴耀斌	张后苏	崔秀梅	彭先定
雷方桂			

前言 一

《计算机语言实用程序与编程技巧》丛书暂包括六个分册。它们是:《C 语言》;《Turbo Pascal 语言》;《80386 汇编语言》;《BASIC 语言》;《FOXBASE+ 数据库语言》;《FOXPro 数据库语言》。

编写这套丛书的目的是为了帮助已经掌握了以上语言基本编程方法的读者进一步提高程序设计技术,学习高级程序设计的方法和技巧。在应用程序设计工作中,借鉴于已有的、经过优化的现有实用程序与实用程序段,使用到应用程序中去,节省编程周期,不断提高应用程序的质量和效果。它是大专生及相关人员最好的教学参考书和工具书。

近年来,各种程序设计语言,从 BASIC 到 C 都在不断的完善和丰富自己,版本不断更新,各种语言在保持自己特色的同时,功能上均在相互接近,互相渗透,取长补短,不断改进。掌握好任意一门程序设计语言都能满足绝大部分应用程序设计的需要。本丛书选材正是从各种语言的最新版本功能出发,叙述各种语言的高级程序设计技术和实用程序。使读者缩短掌握语言新功能的途径。

《C 语言》不仅叙述了基本的静态、动态数据结构,还系统地阐述了内存管理、视频系统及应用、图形、资源辨认、常驻内存程序、数值计算等方面高级程序设计方法,有很多都是作者最新科研成果的总结。

《Turbo Pascal 语言》由于其结构化、模块化特点,不仅是大专学生的基础课程,也是当今中学生奥林匹克竞赛的规定语言之一。本书不仅适合大专学生提高的需要,也适合竞赛需要,适合中学生提高 PASCAL 语言程序能力的参考书。

《80386 汇编语言》则以 8088/8086 为基础,重点阐述 80386、80486 和 80387、80487 的程序设计技术,供程序员参考使用。

《BASIC 语言》是传播最早、最广泛的计算机语言。它对 Quick Basic 和 Ture Basic 功能作了详细叙述,其矩阵语句、图形功能、动画能力、计算能力、I/O 功能及人机图形、窗口界面丝毫不弱于其它任何语言,编程者可以设计出具有各种特色的应用程序。

《FOXBASE+ 数据库语言》总结了目前最常用和实用的程序设计技术,在菜单设计、窗口设计、查询搜索、工具应用等方面都进行了系统的阐述,还有不同类型管理系统的实用程序示例,供读者直接引用或模仿。它是数据库管理系统开发人员和应用程序设计人员最好的一本参考书。

参加本书编写的人员都是长期从事教学和科研工作的教师,他们具有扎实的基础理论和实践能力。全书所有程序均经过上机验证,不少是科研成果的总结汇编而成,写法上注意了先进性、实用性,尽量符合读者学习的思维过程,因而可读性好,极具实用价值。

由于计算机软件的发展一日千里,丛书不可能全部包含当前应用中最新内容,不当之处,尚望批评指正,并愿为广大读者提供咨询服务。

编委会

1994 年 10 月

前 言 二

随着微型计算机在我国的逐步普及,计算机的应用已从科学计算、实时控制方面逐渐扩展到数值处理的各个领域。尤其是微型计算机系统在企事业管理以及办公室自动化的应用方面,正日益受到各部门极大关注。

普及和提高使用微型计算机是计算机事业发展的关键,在数据处理和事务管理使用中微型计算机已日臻成熟,从工资管理、人事管理、仓库管理、设备管理、图书教材管理等都得到广泛的应用。FOXBASE⁺(2.10),由于其高速度功能强的特点更为人们所欢迎。广大读者希望得到一些应用实例,以便学习、参考和借鉴。此书给出几个实际管理系统应用程序。在这些应用程序设计中运用了有关屏幕与菜单、数据录入操作、组合条件查询、报表的打印等设计技巧。同时为了进一步深入学习 FOXBASE⁺的需要,从具体应用实例入手,深入浅出地介绍 FOXBASE⁺编程技巧及 FOXBASE⁺与其它高级语言的相互调用。

本书的管理系统是作者为企事业单位研制的,程序采用 FOXBASE⁺(2.10)编制并全部上机通过,为了便于读者阅读,在应用程序中将数据库的数量及字段项目有些简化了,读者在编制自己的应用系统时,根据实际情况添加,不影响程序实质内容。

全书共七章,第一、二、三、六章由陈福盈编写;第四、五、七章由杨路明编写。中南工业大学王云宜教授担任审校,不少同志在编写本书的过程中给予了大力帮助和指导,特此致谢。

由于编写时间仓促和作者水平有限,书中不当之处在所难免,敬请读者批评和指正。

编者

1994年10月

目 录

1	FOXBASE⁺扩充功能概览	(1)
1.1	FOXBASE ⁺ 组成及运行环境	(1)
1.2	FOXBASE ⁺ 的用户文件	(1)
1.3	内存变量与数组	(3)
1.4	FOXBASE ⁺ 新增加的函数	(4)
1.5	FOXBASE ⁺ 新增加的命令	(5)
2	FOXBASE⁺实用编程技巧	(12)
2.1	菜单设计	(12)
2.2	查询设计	(24)
2.3	报表程序设计	(33)
2.4	自动生成报表的通用程序	(37)
2.5	系统的安全性	(44)
3	工资管理应用程序设计	(50)
3.1	工资管理应用程序的系统结构	(50)
3.2	工资数据库的结构	(51)
3.3	主控模块(GZZK.PRG)	(52)
3.4	增加记录模块(GZSR.PR^)	(54)
3.5	修改所有记录模块(GZXG.PRG)	(57)
3.6	查询、修改、删除模块(GZCSG.PRG)	(58)
3.7	打印工资条模块(DYGZB.PRG)	(62)
3.8	打印工资条模块(DYGZT.PRG)	(65)
3.9	打印工资汇总表模块(DYTOTAL.PRG)	(67)
3.10	计算及打印钞票张数模块(DYXP.PRG)	(70)
4	微型机仓库管理系统	(77)
4.1	系统概况	(77)
4.2	系统设计	(77)
4.3	系统所用数据库的结构	(84)
4.4	程序清单	(89)
5	微型机教材管理系统	(109)
5.1	系统分析	(109)
5.2	系统设计	(109)
5.3	程序清单	(118)
6	设备管理应用程序设计	(142)
6.1	设备管理应用程序的系统结构	(142)
6.2	设备数据库的结构	(143)

6.3	主控模块(SBGL.PRG)	(144)
6.4	数据输入模块(SBFLA.PRG)	(147)
6.5	打印输出模块(SBGLC.PRG)	(150)
7	与系统语言互相调用	(166)
7.1	FOXBASE+程序对外部程序的调用	(166)
7.2	FOXBASE+程序与高级语言程序的间接数据传递	(168)
7.3	FOXBASE+与高级语言程序的直接传递数据	(172)
参考文献		(185)

1 FOXBASE⁺扩充功能概览

美国 FOX SOFTWARE 公司 1987 年推出 FOXBASE⁺关系型数据库管理系统, 该系统与 dBASE III 及 dBASE III PLUS 系统完全兼容, 其平均速度大约是 dBASE III 的六倍, 且在功能上作了许多扩充, 它包含了编译手段, 可以使用 1 维、2 维数组, 提供了过程文件编译器及一些 dBASE III 不具备的功能和函数。对众多的用户来说, 原有的 dBASE III 程序无须作多大修改就能在 FOXBASE⁺上运行, dBASE III 的全部概念、方法、编程技术都可以使用, 且功能速度增加很多。因而 FOXBASE⁺很快在我国推广运行, 是 dBASE III 的更新系统。

1.1 FOXBASE⁺组成及运行环境

FOXBASE⁺有单用户和多用户两种版本, 主要组成文件如下:

单用户版本

FOXPLUS.EXE 执行文件

FOXPLUS.OVL 覆盖文件

FOXPLUS.HLP 帮助文件

FOXBIND.EXE 过程组合程序

FOXPCOMP.EXE 伪编译程序

多用户版本

MFOXPLUS.EXE 执行文件

MFOXPLUS.OVL 覆盖文件

FOXPLUS.HLP 帮助文件

FOXBIND.EXE 过程组合程序

FOXPCOMP.EXE 伪编译程序

单用户版本编写的软件可以在多用户版本下运行, 单用户与多用户版本 100% 地兼容。目前大多用多用户版本 2.0, 2.10。本章以 FOXBASE⁺2.10 为基础进行叙述。

运行 FOXBASE⁺, 操作系统为 MS-DOS 2.0 以上版本。如在网络上使用, 则至少须 DOS3.1 以上版本, 运行 XENIX FOXBASE⁺, 操作系统应为 XENIX 286 3.0 和 SYSTEM V 以上。

硬件至少需要一个软盘驱动器和一个 20M 以上的硬盘驱动器。采用 8087/80287 处理器将能帮助发挥 FOXBASE⁺最大性能。在 80286 机器上运行 DOS, FOXBASE⁺最小的内存要求是 640K。可用 CCDOS 4.0 选驻留一级字库或用 CCDOS2.13 进入汉字状态。一般情况下, FOXBASE⁺将在内存为 1MB 的系统上运行。

FOXBASE⁺系统的启动过程首先是将 FOXBASE 系统盘的五个文件直接拷入硬盘后, 并且要在 CCDOS 所在的目录上建立好 CONFIG.SYS 文件, 在 FOXBASE⁺的目录上建立 CCONFIG.FX 文件。在 CCDOS 系统提示符下键入:

C>MFOXPLUS

即进入圆点提示符(.)状态。

退出 FOXBASE⁺是在圆点提示符时键入 QUIT 命令退回到操作系统状态 C>。

1.2 FOXBASE⁺的用户文件

FOXBASE⁺的用户可生成九类文件来构成自己的应用系统，这九类文件与 dBASE III 的文件相同，但有如下差异：

(1) 可同时打开文件数：在 FOXBASE⁺下，用户可以同时打开文件数最多达 48 个，默认为 16 个，可以同时打开数据库文件为 10 个，每个数据库文件最多可同时打开 7 个索引文件。

(2) 索引文件：FOXBASE⁺中的索引文件默认的扩展名不是 NDX，而是 IDX，它是索引文件第一默认扩展名，第二默认扩展名才是 NDX。当未指定扩展名的索引文件时，首先找 IDX 文件，若找不到则自动找 NDX 文件。

(3) 编译命令文件：FOXBASE⁺可以运行命令文件，但是将它编译后，从磁盘读入文件的速度更快，在编译的同时将文件中不必要的空格和注释删除，压缩原来文件的长度达原来的 30%—50%，即节省磁盘空间，提高程序执行的装载速度。编译的优点还在于它可以在编译的同时检查程序的任何语法错误，且编译后的程序是无法打开修改的，可以起到加密的作用。

编译的方法是在 C>提示符下键入：

```
C>FOXPCOMP <命令文件名 1> [<命令文件名 2>...]
```

注意：命令文件名的扩展名 PRG 必须逐个编译，产生同名而扩展名为 FOX 的目标文件。一次可编译多个文件，文件名中可以采用能配符“?”或“*”。

编译操作示例如下：

【例 1】对 EX1.PRG 文件的编译。

```
C>FOXPCOMP EX1.PRG
```

```
FOXBASE+ Compiler 2.10(c) 1988 Fox Software
```

```
Compiling EX1.PRG contains 1327 bytes and 3 names.
```

【例 2】对 EX1.PRG 及 WX2.PRG 文件同时编译。

```
C>FOXPCOMP EX1.PRG EX2.PRG
```

```
FOXBASE+ Compiler 2.10 (c) 1988 Fox Software
```

```
Compiling EX1.PRG contains 1327 bytes and 3 names.
```

```
Compiling EX12.PRG contains 162 bytes and 1 names.
```

或

```
C>FOXPCOMP EX*.PRG
```

```
FOXBASE+ Compiling 2.10 (c) 1988 Fox Software
```

```
Compiling EX1.PRG contains 1327 bytes and 3 names
```

```
Compiling EX2.PRG contains 162 bytes and 1 names.
```

在编译过程中，如果程序中有错误，则会显示错误语句的行号和错误信息及语句本身，并继续往下编译，编译后，源程序仍然保留，执行目标文件也应用 DO 命令。

```
.DO <命令文件名>
```

如果磁盘有同名的 FOX 文件与 PRG 文件同时存在，系统总是执行最新版本的程序，当源程序经编译产生目标文件后，未修改源程序时，则 DO 命令执行的 FOX。如果编译之后又对源程序进行修改，则 DO 命令执行的是 PRG 文件。

(4) 连接过程文件。FOXBASE⁺提供一个专门连接过程文件的功能，允许将多达 128

个不同的程序结合成为一个过程文件.连接过程文件的方法是:

```
C>FOXBIND <目标文件名> <命令文件名 1> <命令文件名 2>...
```

这里<目标文件名>是希望最后生成的文件名,它的扩展名.PRG会自动加上.<命令文件名 1>、<命令文件名 2>...是希望被连接的一系列过程文件,可以使用通配符“*”和“?”,但它的扩展名必须给出.

【例 3】将 EX1.PRГ 及 EX2.PRГ 命令文件连接成目标文件 NEWEX.PRГ.

```
C>FOXBIND NEWEX EX1.PRГ EX2.PRГ
```

```
FoxBIND Utility Rev 2.00 (c) 1987 Foox Software
```

```
Binding EX1.PRГ
```

```
Binding EX2.PRГ
```

1.3 内存变量与数组

dBASE III 所具有的四种类型内存变量在 FOXBASE⁺中都存在,系统默认可同时存在的内存变量数目为 256 个,还可通过配置文件 CONFIG.FX 设置使范围从 128 到 3600 个.

FOXBASE⁺中增加了一维和二维数组,每个数值存放于一个内存变量之中,其特点是同一数组中存放数据的类型允许不相同.

使用 DIMENSION 命令产生一维或二维内存变量数组,二维数组的次序按行排列.

数组元素可作内存变量使用,同时允许用<数组名>=<表达式>的使用方法,此时表达式的值赋给数组的所有元素.

FOXBASE⁺提供两个命令实现数组与数据库字段之间的数据相互传送.

(一) 数据库字段值给数组元素

命令格式: SCATTER [FIELDS <字段名表>] TO <数组名>

字段名表中的第一个字段赋给第一个数组元素,第二个字段值赋给第二个数组元素,如此类推.如无可选项,则当前记录全部字段的值赋给对应的数组元素(记忆字段除外).并且以字段内容决定数组元素的类型.如果该数组变量未曾定义或虽已定义但定义的数组元素小于字段个数,则系统自动扩充数组元素,使数组元素个数等于字段数,反过来,当字段数小于数组元素个数,余下的数组元素其值不变.此命令对记忆型字段无效.

【例 4】TSGL.DBF 字段值赋给数组 Y.

```
..USE TSGL
```

```
.LIST
```

Record#	书号	书名	作者	出版日期	出版社	定价	册数	借完否	摘要
1	10	数据库	唐福强	01/01/83	华北出版社	2.80	90	.T.	Memo
2	12	计算方法	李 明	02/24/85	华南出版社	3.20	87	.T.	Memo
3	13	计算机原理	唐福强	03/15/86	科学出版社	5.20	100	.F.	Memo
4	14	操作系统	王晓明	02/23/84	华中出版社	3.00	98	.F.	Memo
5	15	汇编语言	王明德	06/12/85	华南出版社	2.90	94	.F.	Memo
6	16	计算机原理	张 明	05/31/86	中南出版社	2.70	94	.T.	Memo

.DIMENSION Y ((3)

.GO 3

SCATTER TO Y

.LIST MEMORY

Y Pub A

(1) N 13 (13.00000000)

(2) C "计算机语言 "

(3) C "唐福强 "

(4) D 03 / 15 / 86

(5) C "科学出版社"

(6) N 5.20 (5.20000000)

(7) N 100 (100.00000000)

(8) L .F.

1 variables defined 55 bytes used

255 variables available, 5945 bytes available

(二) 数组元素替换数据库字段值

命令格式: GATHER FROM <数组名> [FIELDS<字段名表>]

该命令将数组名元素的值——替换字段名表中对应的字段值。如果省略可选项, 则替换所有字段。若字段小于数组元素的个数, 则仅仅替换前面的字段, 反之则余下的元素将被忽略, 但是数组元素的类型必须符合相应字段的类型, 否则会出现“DATA TYPE MISMATCH”(数据类型错配)的出错信息。

【例 5】Y (2), Y (3) 元素值替换书名、作者字段值。

.DIMEENSION Y(8)

.USE TSGL

.Y(2)="人工智能"

.Y(3)="王维铭"

.GATHER FROM Y

.DISPLAY

记录号	书名	作者	出版日期	出版社	定价	册数	借完否	摘要
-----	----	----	------	-----	----	----	-----	----

3	13 人工智能	王维铭	03 / 15 / 886	科学出版社	2.90	100	.T.	Memo
---	---------	-----	---------------	-------	------	-----	-----	------

• Y(4)='10 / 01 / 91'

.GATHER FROM Y

Data type mismatch.

1.4 FOXBASE⁺ 新增加的函数

FOXBASE⁺包含了 dBASE III 的全部函数, 有些函数的功能有所扩展, 同时还增加了一些新的函数, 本节按其用途分类介绍其中一些有用的函数。

(1) 重复字符串函数。

格式: REPLICATE (<字符串表达式>, <数值表达式>)

功能: 重复<字符串表达式>的内容, 重复次数由<数值表达式>的值决定。

例: @ 2, 3 SAY REPLICATE ("*", 10)

为在屏幕上 2 行 3 列的位置连续显示 10 个 "*" 号。

(2) 出错信息函数。出错信息函数有两个, 它们的格式分别为 ERROR () 和 MESSAGE ()

ERROR () 函数测试程序运行中最近出现的一次错误, 并返回一个错误号, 如无出错则返回 0 值, 用 MESSAGE () 函数显示相应出错号的出错信息, 它说明出了什么性质的错误。

注意: 这两个函数只有被 ON ERROR 命令激活时才能测试出错误号及返回错误信息字符串。

【例 6】图书管理数据库文件中的某一字段名为“书名”, 但在编写程序时错写为“书命”, 因而, 测试出错误号及出错信息。

```
TYPE EX8.PRG
```

```
ON ERROR DO ERR
```

```
USE TSKL
```

```
@ 3, 10 SAY 书命
```

```
RETURN
```

```
.TYPE ERR.PRG
```

```
@ 5, 100 SAY '出错号: '+STR(ERROR( ), 4)
```

```
@ 7, 10 SAY '出错信息: '+MESSAGE( )
```

```
RETURN
```

当执行 EX8.PRG 命令文件时

```
DO EX8
```

出错号: 12

出错信息: Variable not found

(3) 求字段名函数。

格式: FIELD (<数值表达式 1> [, <数值表达式 2>])

功能: 该函数返回指定工作区中打开库文件中指定序号的字段名, 结果为字符串, 如无对应的字段, 则为空串。

其中: <数值表达式 1> 的值为字段序号。<数值表达式 2> 为工作区号, 缺省时为当前工作区。

【例 7】某一数据库的结构如下:

Field	Field Name	Type	Width	Dec
1	书号	Numeric	5	
2	书名	Character	16	
3	作者	Character	8	
4	出版日期	Data	8	
5	出版社	Character	10	
6	定价	Numeric	5	2

7	册数	Numeric	5
8	借完否	Logical	1
9	摘要	Memo	10

?FIELD (5)

出版社

?FIELD(10)

(4) 动态按键函数。

格式: INKEY ([数值表达式])

功能: 该函数给出一个整数值, 这个数值是用户执行程序过程中键入的键的 ASCII 码值, 其值在 0-255 之间, 如果用户什么键也不按, 则返回 0 值, 如果键盘缓冲区有几个字符, 则返回第一个被键入的字符的 ASCII 码值。

函数中的数值表达式是指定等待用户按键时间(单位为秒)。若数值表达式值为 0, 则无限期等待, 直到击键为止。

INKEY () 函数赋予我们对某个系统的控制权, 任何时候都可以知道用户按下的是什么键。

(5) 自定义函数。FOXBASE⁺允许用户定义特殊功能的函数。

自定义函数对应一个命令文件, 该命令文件的结构如下:

PARAMETER <参数表>

函数体

RETURN <函数值>

自定义的命令函数以参数语句 PARAMETER 开头, 其后参数表是一个内存变量, 相互间用逗号隔开, 函数值是通过 RETURN 命令返回。函数调用时只需要引用“自定义函数”相应的命令文件名, 括号中填入调用参数即可。注意: 调用参数要与函数命令文件中的参数在个数和类型上一一对应。

【例 8】FOXBASE⁺有一个月份函数 CMONTH (), 这个函数根据给出的日期求出其当时的月份, 并用英文显示出来。

现在用户自定义一个汉字月份函数, 其使用与 CMONTH () 完全一样, 只是其函数值换成了汉字。定义求汉字月份函数的程序如下:

*****CCMONTH.PRG*****

PARAMETER NUM

Y=MONTH (NUM)

DO CASE

CASE Y=1

M='元'

CASE Y=2

M='二'

CASE Y=3

M='三'

CASE Y=4

M='四'

```

CASE Y=5
    M='五'
CASE Y=6
    M='六'
CASE Y=7
    M='七'
CASE Y=8
    M='八'
CASE Y=9
    M='九'
CASE Y=10
    M='十'
CASE Y=11
    M='十一'
CASE Y=12
    M='十二'
ENDCASE
M=M+'月'
RETURN N

```

设有一日期型变量，其值为 10/01/94。

```
MDY=CTOD('10/01/94')
```

使用 FOXBASE⁺提供的求月份函数：

```
CMONTH (MDY)
```

```
October
```

使用自定义函数求月份函数：

```
?CCMOTH (MDY)
```

```
十月
```

1.5 FOXBASE⁺新增加的命令

FOXBASE⁺包含汉字 dBASE III 全部命令及其全部有功能，很多命令的功能有了新的扩充，并且增加了一些新的有用的命令，本节只叙述其中的一部分。

(一) FOXBASE⁺命令的一般形式

命令动词 [<范围>] [<表达式>] [FOR<条件>] [WHILE<条件>]

其中：WHILE 子句和 FOR 子句可同时使用，但在处理时 WHILE 优先。

范围项增加一个选择项 REST。它表示范围从当前记录到文件尾的所有记录。

其余子项与汉字 dBASE III 完全相同。

(二) 功能扩展及新增加的命令

(1) 屏幕输入输出命令。

@ <行, 列> [SAY <表达式> [PICTURE <子句 1>] [GET< 变量> [PICTURE <子句 2>] [RANGE <数值表达式 1>, <数值表达式 2>]] [VALID <条件表达式>]]

①如果有选择项 PICTURE<子句 1>, 则<表达式>按子句 1 的方式输出。

②如果有选择项 PICTURE<子句 2>, 则输入的格式受到<子句 2>的控制。

③RANGE 选择项指数值型变量和日期型变量上下界, 其数值表达式用于限制 GET 响应所输入的数据的最大值与最小值。其类型必须与 GET 给出的变量相同, 如果是日期型常量、要用 CTOD () 函数转换。

④VALID 子句用来限制和核实 GET 变量的输入值, 其功能比 RANGE 子句更强, 当执行 READ 语句, 在为 GET 子句的逻辑表达式进行审查, 若表达式为.T. (真) 时, 输入数据被承认, 若表达式为.F. (假) 时, 输入数据不被承认。系统提示按空格键后重新输入。

VALID 子句的<逻辑表达式>可根据 GET 变量由用户输入的值进行计算, 也可以包含用户自定义函数。因此检查功能较强。

例: 选择项只允许输入一位数字型数据, 其它输入无效。

OP1=0

@ 2, 0 SAY "请选择" GET OP1 PICT "9" RANG 0, 2

READ

例: 选择项只允许输入一位逻辑型数据, 其它输入无效。

OP2=.T.

@ 2, 0 SAY "请选择" GET OP2 PICT "L"

READ

例: 在 VALID 子句中的<逻辑表达式>根据 GET 变量 (假设为设备编号) 由用户输入的值进行计算逻辑值, 以决定数据输入的有效性。

@ 2, 0 SAY "设备编号" GET 设备编号;

VALID (设备编号>100 .AND. 设备编号<1000)

READ

(2) 画方框命令。

①@ <行, 列> [CLEAR]TO <行 2, 列 2> [DOUBLE]

此命令可以在屏幕上画出方框, 命令中<行 1, 列 1>给出方框左上角的行列坐标, <行 2, 列 2>给出方框右下角的行列坐标, 有 DOUBLE 可选项时, 则方框边线为双线, 否则为单线, 有 CLEAR 可选项时则为擦除方框。

②@ <行 1, 列 1, 行 2, 列 2> BOX [<字符串表达式>]

此命令在指定的左上角和右下角的坐标上用字符串表达式中指定的字符画一个方框。字符串表达式最多由 9 个字符组成, 其中第 1-8 字符依次用于方框的左上角、上边线、右上角、右边线、右下角、下边线、左边线, 而第 9 个字符用来填充方框内。

若省略字符串表达式, 则一个单线框, 如果字符串表达式仅一个字符, 就用此字符画方框。例如:

@ 1, 2, 4, 10 BOX 'ABCDEFGH' @ 1, 2, 4, 10 BOX ' * '


```

ABBBBBBBC          * * * * *
H          D        *          *
H          D        *          *
GFFFFFFFE          * * * * *

```

(3) 菜单选择命令。传统式的菜单选择是由@...SAY语句组成功能选择菜单，由GET给一个变量供用户选择。

FOXBASE⁺提供另一菜单选择功能，其命令格式如下：

@ <行, 列> PROMPT <字符串表达式 1> [MESSAGE <字符串表达式 2>] 和 MENU TO <内存变量名>

这两个命令是联合使用的，@...PROMPT的功能相当dBASE III 中@...SAY，它由行, 列指定的位置显示<字符串表达式 1>的值，可以有多条此语句，然后由MENU TO命令激活，并且在菜单选择项上，以高亮度的光亮块指明。用户可以用光标控制键移动选择。另一种选择方法是键入<字符串表达式 1>中的项目首字符，此时<内存变量名>返回一个数字型的整数，系统判断返回的值刚好与这组@...PROMPT命令等程序时的次序相等时，转向相应的子程序处理。

选择项[MESSAGE<字符串表达式 2>]是对<字符串表达式 1>的内容进行解释。

(4) 出错命令。

格式: ON ERROR [<命令>]

设置此命令后，当程序执行出错时，系统转去执行ON ERROR 后面的命令。此命令通常是调用一个错误处理子程序即DO<命令文件名>。

而ON ERROR命令使ON ERROR <命令>不起作用。

(5) 击Esc键处理命令。

格式: ON ESCAPE [<命令>]

通常运行程序时，忌讳按Esc键来强行中断程序的执行，因为有可能对打开的数据库文件及其索引文件造成破坏，引起数据丢失，为此在主程序的开头设置一条命令：

ON ESCAPE DO ESCPROC

即可避免此弊病。

ESCPROC.PRG是Esc键处理程序，当主程序在执行过程中，不管何时，只要击Esc键时，主程序将转向去执行ON ESCAPE后面的处理程序ESCPRO.PRG，该处理程序将发出警告，待稍后几秒后再返回主程序继续执行避免了人为强迫中断程序引起的不良后果。

【例9】

```

* * * * EX11.PRG * * * * *
SET TALK OFF
ON ESCAPE DO ESCPROC &&程序转向 ESCPROC.PRG 执行
WAIT '请试击 Esc 键中断运行' TO ZD
@ 5, 10 SAY "请记住: 系统的运行时不得随意击 Esc 键!"
ON ESCAPE &&关闭 Esc 键击活功能
RETURN

```