

FoxBASE+通用程序设计方法 与技巧

徐尔贵 著



73.8741 73.874
C62 C62

6114-12
932117

图 3 清

FoxBASE⁺通用程序设计 方法与技巧

徐 尔 贵 著

国防工业出版社

· 北京 ·

(京)新登字 106 号

图书在版编目(CIP)数据

FoxBASE⁺通用程序设计方法与技巧/徐尔贵著. —北京
国防工业出版社, 1995. 4
ISBN 7-118-01377-3

I. F... II. 徐... III. FoxBASE 语言-程序设计-基本知识
IV. TP312Fox

中国版本图书馆 CIP 数据核字(94)第 12832 号

国防工业出版社 出版发行

(北京市海淀区紫竹院南路 23 号)

(邮政编码 100044)

北京大兴兴达印刷厂印装

新华书店经售

*

开本 787×1092 1/16 印张 19 437 千字

1995 年 4 月第 1 版 1995 年 4 月北京第 1 次印刷

印数 1—5000 册

定价: 20.00 元

(本书如有印装错误, 我社负责调换)

前 言

FoxBASE⁺关系数据库管理系统是美国 Fox Software 公司推出的最新软件,深受广大用户的欢迎和瞩目。

FoxBASE⁺既可用于多用户系统,也可用于单用户系统,并与 dBASE III 和 dBASE III plus 完全兼容,也就是说用 dBASE III 或 dBASE III plus 编写的源程序不经任何修改就可以在 FoxBASE⁺环境下运行。然而 FoxBASE⁺与 dBASE III 相比,在功能上有许多扩充和增强,如增加了许多命令、函数、数组和记录锁定保护措施等;在运算速度上比 dBASE III 快近 6 倍。此外,FoxBASE⁺对运行环境的适应面很宽,在硬件环境方面从 IBM/PC 到 VAX11/780,在软件环境方面从 MS-DOS 到 UNIX 都可适用。

鉴于 FoxBASE⁺的优越性,近来许多用户和开发单位都采用 FoxBASE⁺开发管理软件。有些读者和用户已熟悉 dBASE III,并且能够开发某些专用管理程序,但仍希望掌握先进的数据库管理系统,进一步提高程序设计水平,以期达到能够开发商品管理软件的水平。目前出版的 FoxBASE⁺方面的书籍虽然不少,但有关通用程序设计方法与技巧的书仅国防工业出版社《FoxBASE⁺原理及其通用程序设计方法与技巧》一本。这本书问世以后,很受读者欢迎,其中读者最感兴趣的是通用程序设计部分,这也正是该书的独到之处。为此,又写了现在的《FoxBASE⁺通用程序设计方法与技巧》,它去掉了原书的原理部分,又对通用程序设计部分做了修改。本书有如下 4 个特点。

(1)用清晰的概念和实例介绍了 FoxBASE⁺对 dBASE III 命令与函数功能扩充,从而为已熟悉 dBASE III 的读者和用户迅速掌握 FoxBASE⁺提供方便和捷径。

(2)系统地提出、归纳和论述通用管理程序设计 10 项方法和技巧,其中每种方法和技巧都附有实例程序。这些方法和技巧都围绕解决通用管理程序设计中遇到的三个基本问题:允许用户定义库文件结构、生成用户随机选定的字段名表和生成用户随机选定的条件逻辑表达式。前者解决通用程序设计中的库文件结构的随机性问题,后者是将 FoxBASE⁺命令中的两个最核心的子句——字段名表子句和条件表达式子句由特殊集扩展为一般集,方法独特、方便而实用。

(3)论述带有状态记忆的窗口式菜单程序设计的方法与艺术,这对于方便操作和装饰商品软件是必不可少的。

(4)通过通用程序设计实例的介绍,进一步掌握通用管理程序设计方法和技巧。

全书包括五章:第一章介绍 FoxBASE⁺对 dBASE III 命令和函数功能扩充;第二章介绍通用管理程序设计原则及常用语句和函数。第三章论述通用管理程序设计方法和技巧;第四章以实用、美观的各类窗口式菜单实例介绍菜单程序设计方法、技巧与艺术;第五章以通用工资管理系统设计为例,综合介绍通用管理程序设计方法与技巧的运用。

本书在内容安排上由浅入深,循序渐进。既适用于做大专院校教材,也适用于计算机

用户和程序设计人员学习和参考。

为配合本书学习,使您尽快掌握 FoxBASE⁺ 通用程序的设计方法和技巧,设计出高质量的商品管理软件,另发行通用程序设计学习盘一张。本盘包括两部分内容:一部分是通用程序设计方法与技巧的论述和说明,为方便使用,这部分内容既可打印,也可显示;另一部分内容是通用程序设计方法使用练习。用户使用本盘会极其方便地练习通用程序设计方法与技巧,既可以执行标准示例,也可执行用户定义条件和模式。此外本盘还包括各类带状态记忆窗口式菜单程序和通用工资管理系统。为便于学习,本盘一律以源程序形式给出,并设有总控程序和标准数据恢复程序。该盘既可用做教学盘,也可用做自学盘。

作者 徐尔贵

目 录

第一章 FoxBASE⁺组成及对 dBASE III 功能扩充	1
§ 1.1 FoxBASE ⁺ 的组成和启动	1
一、FoxBASE ⁺ 的组成	1
二、FoxBASE ⁺ 的启动	2
§ 1.2 函数扩充	2
§ 1.3 命令扩充	7
第二章 通用管理程序设计原则及常用语句和函数	12
§ 2.1 管理程序通用化的意义与设计原则	12
一、管理程序通用化的意义	12
二、通用管理程序设计原则	13
§ 2.2 通用程序设计中常用的 FoxBASE ⁺ 语句和函数	15
一、COPY 语句	15
二、CREATE 语句	17
三、FCOUNT 函数	17
四、FIELD 函数	18
第三章 FoxBASE⁺通用管理程序设计方法与技巧	19
§ 3.1 库文件结构的测试	19
一、用 COPY 命令测试库文件结构	19
二、用 FCOUNT 和 FIELD 函数测试库文件结构	20
§ 3.2 程序方式建立库文件结构	21
§ 3.3 程序方式修改库文件结构	23
§ 3.4 任一变量组的屏幕显示格式安排	24
§ 3.5 生成随机选定的字段名表	25
§ 3.6 生成随机组合的逻辑表达式	30
§ 3.7 模式方式下的数据核算	36
§ 3.8 表格化核算方法与实参文件	44
一、什么是表格化核算	44
二、表格化核算类型	45
三、表格化核算方法的通用化与实参文件	47
四、建立实参文件程序	51
五、确定表格运算顺序	52

六、表格核算程序	54
七、表格化核算实例——简单成本核算	62
§ 3.9 模式方式采集表格数据	65
一、建立数据采集模式文件	65
二、报表与表格数据采集	70
§ 3.10 报表格式数据的生成与数据装配打印	71
一、建立报表栏目程序	72
二、建立报表标题和表尾内容程序	86
三、报表打印程序	89
第四章 菜单程序设计技巧与艺术	96
§ 4.1 设计窗口式菜单程序常用语句和函数	96
§ 4.2 窗口式菜单程序设计技巧与艺术	98
§ 4.3 二级窗口式菜单程序设计(一)	110
§ 4.4 二级窗口式菜单程序设计(二)	116
§ 4.5 三级窗口式菜单程序设计	129
第五章 FoxBASE+通用工资管理系统程序设计	131
§ 5.1 通用工资系统及主控模块	131
一、通用工资管理系统的基本结构	131
二、主控模块	132
§ 5.2 工资系统初始化	151
一、填写工资款项	151
二、填写部门名称	153
三、填写人员类别	154
四、填写工资计算模式	154
五、建立工资分录模式	156
六、建立工资分录所需会计科目	157
§ 5.3 工资系统数据录入	176
§ 5.4 工资核算汇总	179
一、当月工资数据核算	179
二、部门小组工资汇总	180
三、工资面额统计	180
四、当年工资累计	181
§ 5.5 工资系统查询	200
一、当月工资查询	200
二、当年综合查询	200
三、月度工资档案查询	200
四、年度工资档案查询	201

§ 5.6 工资数据报表输出	219
一、打印部门工资汇总表	219
二、打印部门小组工资汇总表	219
三、打印小组工资表	219
四、打印工资条	220
五、打印工资面额张数表	220
六、工资数据存档	220
§ 5.7 工资帐务分录	239
一、工资分录核算	240
二、打印工资记帐凭证	240
§ 5.8 工资系统维护	244
§ 5.9 工资数据维护	244
一、修改月度数据	244
二、一般选择修改	245
三、删除人员数据记录	245
四、增加人员数据记录	245
五、工资款项统一置数	245
六、工资款项统一增数	246
§ 5.10 工资款项类别维护	270
一、增减工资款项	270
二、修改工资款项	270
三、增减部门小组	270
四、增减人员类别	270
五、工资数据排序	271
§ 5.11 工资计算模式维护	280
附录一 FoxBASE⁺命令表	287
附录二 FoxBASE⁺函数表	292

第一章 FoxBASE⁺组成及对 dBASE III 功能扩充

FoxBASE⁺与 dBASE III 完全兼容,也就是说 dBASE III 程序不做任何修改即可在 FoxBASE⁺下运行。然而 FoxBASE⁺新增许多函数和命令,并对原 dBASE III 命令功能做了许多扩充。读者只要掌握本章内容,就可以使用 FoxBASE⁺设计一般应用程序,若再掌握本书介绍的通用程序设计方法与技巧,就具备了设计通用管理软件的能力。

下面首先介绍 FoxBASE⁺的组成,然后介绍扩充的某些常用函数,最后介绍扩充的某些常用命令。

§ 1.1 FoxBASE⁺的组成和启动

一、FoxBASE⁺的组成

FoxBASE⁺由以下文件组成:

1. MFOXPLUS. EXE 或 FOXPLUS. EXE

该文件是 FoxBASE⁺的总控模块并常驻内存,是一个可执行文件。文件名中以 M 字符打头的是多用户版本,而没有 M 字符打头的文件是单用户版本。多用户版本可用于单用户系统,而单用户版本却不能用于多用户系统。

2. MFOXPLUS. OVL 或 FOXPLUS. OVL

这个文件是 FoxBASE⁺的可覆盖模块,负责解释和编译执行各种 FoxBASE⁺命令。

3. FOXPHHELP. HLP

它是 HELP 命令使用的文件,帮助用户了解和掌握 FoxBASE⁺的各种命令和函数的语法、功能和使用方法。

4. FOXBIND. EXE

该程序是过程文件编辑器,用户使用这个文件可以极其方便地将其指定的各命令文件自动地组成过程文件。

5. FOXPCOMP. EXE

该文件是 FoxBASE⁺的编译程序,可使用编译程序对用户源程序进行编译,生成短小精悍、执行速度较快,且可在解释程序下执行的代码。FoxBASE⁺的编译属“伪编译”,不能生成 DOS 可执行代码,而是生成所谓的“Token”化代码。

以上 5 个文件占用磁盘容量如下:

MFOXPLUS. EXE	247808 字节
MFOXPLUS. OVL	138032 字节
FOXPHHELP. HLP	149909 字节
FOXBIND. EXE	26119 字节

FOXPCOMP. EXE

72480 字节

其中, MFOXPLUS. EXE 和 MFOXPLUS. OVL 两个文件是使用 FoxBASE⁺ 必不可少的。

二、FoxBASE⁺ 的启动

如果计算机配有硬磁盘, 使用 FoxBASE⁺ 是很方便的, 执行速度也较快。在启动吴晓君汉字操作系统 CCDOS2. 13H 以上版本后, 最好先将 FoxBASE⁺ 的 5 个文件用 DOS 的复制命令装入硬盘的某个子目录下, 然后键入

```
>MFOXPLUS
```

或

```
>FOXPLUS
```

并按回车键, 即可进入 FoxBASE⁺ 的圆点提示符下, 准备接收 FoxBASE⁺ 的任何命令。

§ 1.2 函数扩充

1. 测文件结构字段数函数 FCOUNT

格式: FCOUNT([<数值表达式>])

该函数返回由<数值表达式>值指定的工作区中数据文件字段个数。表达式取值 1~10, 若缺省选择项, 则指当前工作区。如果在选择的工作区中没有打开的文件, 则返回值为 0。

例 1-1 测文件结构字段数。

```
. sele 1
. use gzsjk
. sele 2
. use hmc
. ? fcount()
      9
. ? fcount(1)
      45
. ? fcount(3)
      0
```

2. 测文件记录数函数 RECCOUNT

格式: RECCOUNT([<数值表达式>])

该函数返回由<数值表达式>值所指定的工作区中数据文件的记录数。表达式取值 1~10, 若缺省选择项, 则指当前工作区, 若在选择的工作区中没有打开的文件, 则返回值为 0。

注意: 返回值是文件的全部记录条数, 不考虑记录是否打有删除标志, 也不考虑是否经过过滤。

例 1-2 测文件记录数。

```
. use gzsjk
. ? reccount()
```

```

      8
.   go 5
.   delete
.   ? reccount()
      8
.   set filter to 性别="女"
.   ? reccount()
      8
.   sele 2
.   ? recc()
      0

```

3. 键值函数 INKEY

格式: INKEY([<数值表达式>])

该函数返回用户当前所敲键的 ASCII 码值,取值范围在 0~255 之间。<数值表达式>值表示程序等待的秒数,若缺省选择项,则立即返回;若表达式值为 0,则程序无限等待用户敲键。

在设计目前十分流行的窗口菜单程序时常使用这个函数,下面列出窗口菜单程序中常用键及其 ASCII 值。

键	ASCII 码
↑	5
↓	24
→	4
←	19
Esc	27
Enter	13

4. READKEY 函数

格式: READKEY()

功能: 该函数返回一个整数值,它在全屏幕命令退出时测试用户是否经过了修改,且记录了全屏编辑时最后使用的光标控制键。如果数据没有修改,则返回值在 0~36 之间;如果数据进行了修改,则返回值在 256~292 之间,也就是说,它们之间相差 256。全屏命令包括 APPEND、BROWSE、CHANGE、CREATE、EDIT、READ、MODIFY。

击入键、返回值和击入键的功能见表 1-1。

READKEY 函数的应用请参考 § 3.10 报表格式数据生成与数据装配打印一节中建栏目程序。

5. 测文件字段名函数 FIELD

格式: FIELD(<数值表达式 1>[,<数值表达式 2>])

该函数返回<数值表达式 2>指定工作区文件中由<数值表达式 1>值所指定序号的字段名。如缺省选择项则指当前工作区。若在指定的工作区中没有打开的文件,或<数值表达式 1>涉及到一个不存在的字段,则返回值为空串。

表 1-1 键的返回值及功能

按 键	等价键	退出键	修改码	作 用
Ctrl-H	Backspace	0		后退一个字符
Ctrl-S	←	0		后退一个字符
Ctrl-D	→	1	257	前进一个字符
Ctrl-L		1	257	前进一个字符
Ctrl-A	Home	2	258	后退一个字符
Ctrl-F	End	3	259	前进一个字
Ctrl-E	↑	4	260	后退一个字段
Ctrl-K		4	260	后退一个字段
Ctrl-X	↓	5	261	前进一个字段
Ctrl-J		5	261	前进一个字段
Ctrl-R	PgUp	6	262	后退一屏
Ctrl-C	PgDn	7	263	前进一屏
Ctrl-Z	Ctrl-←	8	264	向左浏览
Ctrl-B	Ctrl-→	9	265	向右浏览
Ctrl-U		10	266	删除一个记录
Ctrl-N		11	267	插入一个记录
Ctrl-Q	Esc	12	268	不存盘退出
Ctrl-W	Ctrl-End	14	270	存盘后退出
Ctrl-M	(超过末尾处) (起始处)	15	271	满,超过结束位置
		16	272	APPEND
		16	272	MODIFY STRUCTURE
		16	272	MODIFY REPORT
Ctrl-]	Ctrl-Home	33	289	菜单显示开关
Ctrl--	Ctrl-PgUp	34	290	移出一个记录
Ctrl-^	Ctrl-PgDn	35	291	移入一个记录
F1		36	292	Help 功能键
Ctrl-G	Del	N/A	N/A	N/A
Ctrl-V	Ins	N/A	N/A	N/A

例 1-3 测文件字段数和字段名。

```
. use hmc
```

```
. ? recc()
```

```
10
```

```
. ? field(5)
```

```
婚否
```

```
. ? field(12)
```

```
(返回值为空串)
```

```
. x=field(12)
```

```
. ? type("x")
```

```
C
```

```
. ? len(x)
```

```
0
```

```
. ? field(3,2)
```

(返回值为空串)

6. 记录搜索函数 FOUND

格式: FOUND([<数值表达式>])

该函数返回一个逻辑值,如果离包括该函数的命令最近的命令: CONTINUE、FIND、LOCATE 或 SEEK 执行成功,则函数返回逻辑值“真”;否则,如果搜索命令执行不成功或记录指针被非搜索命令(例如 GOTO 命令)移动,则返回值为“假”。

例 1-4 搜索 hmc 文件中未婚的所有记录。

```
. use hmc
```

```
. list
```

Record #	编号	姓名	年龄	性别	婚否	何时工作	技术职称	工资
1	0101	张金山	32	男	.T.	09/07/80	工程师	105.00
2	0201	何凯	50	女	.T.	09/01/65	高级工程师	140.00
3	0301	张立岩	35	男	.F.	09/01/80	工程师	110.00
4	0401	刘喜旺	58	男	.T.	11/01/50	机师	140.00
5	0501	王珊珊	37	女	.F.	09/01/71	7 级焊工	105.00
6	0502	刘崧岭	45	女	.T.	09/01/71	高级工程师	140.00
7	0601	赵立本	22	女	.F.	09/01/91	技术员	85.00
8	0701	徐海波	34	男	.T.	01/01/81	工程师	105.00
9	0801	冯大千	54	男	.T.	10/01/60	高级工程师	150.00
10	0901	周力	35	男	.T.	10/01/76	6 级电工	100.00

```
. locate for . not. 婚否
```

```
Record=3
```

```
. ? found()
```

```
.T.
```

```
. continue
```

```
Record=5
```

```
. ? found()
```

```
.T.
```

```
. continue
```

```
Record=7
```

```
. ? found()
```

```
.T.
```

```
. continue
```

```
End of locate scope
```

```
. ? found()
```

```
.F.
```

7. 条件函数 IIF

格式: IIF(<逻辑表达式>, <表达式 1>, <表达式 2>)

若<逻辑表达式>值为“真”, 该函数返回<表达式 1>值, 否则返回<表达式 2>的值。

例 1-5 IIF 和 FOUND 函数应用。

```
.  xz=10
.  xz0=iif(xz>9,str(xz,2),str(xz,1))
.  ? xz0
      10
.  xz=9
.  xz0=iif(xz>9,str(xz,2),str(xz,1))
.  ? xz0
      9
.  use hmc
.  loca for 姓名="何凯"
.  xx=iif(found(),技术职称,"无此人名")
?  xx
高级工程师
.  xx=iif(found(),年龄,"无此人名")
.  ? xx
      50
.  loca for 姓名="何凯丽"
.  ? xx
无此人名
```

8. 用户自定义函数

虽然 FoxBASE⁺ 定义近 100 个内部函数, 但有时用户仍希望在程序中自己定义某些特殊函数以简化运算, 这些由用户定义的函数就被称之为“用户自定义函数”——UDF。

其实, UDF 是一段 FoxBASE⁺ 程序, 它包括一组有效的 FoxBASE⁺ 命令, 并将返回值交给调用程序。

UDF 和普通程序一样, 既可用于过程文件, 也可被编译。它既可作为程序被 DO 命令调用, 也可作为函数成分被调用。在作函数调用时, 自定义函数必须返回一个值给调用程序, 否则将产生语法错误。

UDF 作函数调用时的格式是:

<用户自定义函数名>([<参数>])

如果有选择项<参数>, 则 UDF 这个程序的第一条语句必须是参数语句 **PARAMETER**。

可引用用户自定义函数的语句有:

STORE

REPLACE

IF、WHILE、CASE 等控制语句

?, ??

DISPLAY、LIST 等显示列表语句

@...GET...

例 1-6 函数方式和过程方式调用用户自定义函数。

调用自定义函数程序 exam.prg 如下：

```
set talk off
set date ansi
* 函数方式调用自定义函数
accept "请输入日期：" to rq
? hzrqcl(rq)
* 过程方式调用自定义函数
accept "请输入日期：" to rq1
hzrq=spac(14)
do hzrqcl with rq1
? hzrq
retu
```

用户自定义函数程序 hzrqcl.prg 如下：

```
para x
hzrq=subs(x,1,2)+"年"+subs(x,4,2)+"月"+subs(x,7,2)+"日"
retu hzrq
```

程序执行结果如下：

```
.do exam
请输入日期:91.08.01
91年08月01日
请输入日期:91.08.02
91年08月02日
```

程序的功能是：用户输入字符日期数据后，调用自定义函数程序将其转换成汉字日期。在调用程序中安排了函数调用方式和过程调用方式。

语句 set date ansi 的功能是置日期格式为美国国家标准协会制定的日期格式，即“年.月.日”格式，接近中国日期格式。

§ 1.3 命令扩充

FoxBASE⁺对 dBASE III 扩充的常用命令如下：

1. 数据库间建立多个关联

dBASE III 允许两个数据库之间建立关联，而 FoxBASE⁺允许建立多个数据库之间的关联。命令格式如下：

SET RELATION TO [<表达式> INTO <别名>][ADDITIVE]

选择项 ADDITIVE 允许在保留已建关联情况下建立另一个关联。

例 1-7 将学生档案文件 xsda.dbf 与两个学习成绩档案文件 xxcjda1.dbf 和 xxcjda2.dbf 建立关联并显示所需记录内容。

例 1-7

. use xsda

. list

Record #	学号	姓名	年龄	籍贯
1	1001	张金山	18	北京
2	1002	李文海	19	上海
3	1003	赵丽蓉	20	天津
4	1004	王礼仁	21	广州

. use xxcjda1

. list

Record #	学号	数据库	数据结构	编译原理
1	1001	80	90	100
2	1002	70	80	90
3	1003	60	70	80
4	1004	50	60	70

. use xxcjda2

. list

Record #	学号	信息系统	软件工程	模型与模拟
1	1001	100	90	80
2	1002	90	80	70
3	1003	80	70	60
4	1004	70	60	50

. sele 2

. use xxcjda1 index xxcjda1 alias cj1

. sele 3

. use xxcjda2 index xxcjda2 alias cj2

. sele 1

. use xsda

. set relation to 学号 into cj1

. set relation to 学号 into cj2 additive

. display 学号,姓名,cj1->数据库,cj2->软件工程

Record #	学号	姓名	CJ1->数据库	CJ2->软件工程
1	1001	张金山	80	90

. go 3

. display 学号,姓名,cj1->数据库,cj2->软件工程

Record #	学号	姓名	CJ1->数据库	CJ2->软件工程
3	1003	赵丽蓉	60	70

. ? recn()

3

. sele 2

. ? recn()

3

```
. sele 3
. ? recn()
3
```

2. 建立内存变量数组

格式: **DIMENSION** <内存变量>(<数值表达式1>[,<数值表达式2>])
 [,<内存变量>(<数值表达式1>[,<数值表达式2>])...]

DIMENSION 命令用于定义一维或二维内存变量数组。内存变量数组的起始下标为1, 数组元素的类型在定义时为逻辑型, 并赋以逻辑值假.F., 其后可用赋值语句改变数组元素的类型和值。

数组规模既受内存变量总数限制, 也受内存变量存储空间限制。

二维数组元素既可用双下标引用, 也可用单下标引用。例如对一个2*3的数组x, 双下标表示数组元素为:x(1,1)、x(1,2)、x(1,3)、x(2,1)、x(2,2)和x(2,3); 而用单下标表示数组元素为:x(1)、x(2)、x(3)、x(4)、x(5)和x(6)。

例1-8 定义一维数组x(4)和二维数组y(2,3)。

```
. dimension x(4)
. dimension y(2,3)
. list memo

X          pub      A
( 1)      L.F.
( 2)      L.F.
( 3)      L.F.
( 4)      L.F.

Y          pub      A
( 1, 1)   L.F.
( 1, 2)   L.F.
( 1, 3)   L.F.
( 2, 1)   L.F.
( 2, 2)   L.F.
( 2, 3)   L.F.
```

3. 数据库向数组传递数据

格式: **SCATTER** [FIELDS<字段名表>] TO <数组>

执行该命令, 从数据库文件当前记录的第一个字段起, 依次将各字段内容传送给数组各元素, 数组元素的数据类型随之也取相应字段的数据类型。如果命令中缺省字段名表, 则除备注型字段外将传送所有字段。若数组元素的数目多于被传送的字段数目, 则数组的多余元素保持不变。倘若指定的数组不存在, 或者该数组的大小容纳不下全部被传送的字段, 则系统会自动定义一个新数组。

例1-9 将cbd.dbf文件的末记录内容传送给变量数组。

```
. use cbd
. list

Re# 部门      材料费      燃料费      工资  车间经费  企业管理费  车间成本
1 一车间  30000.00  10000.00  5000.00  5000.00   2000.00  52000.00
```