

dBASE III PLUS

GUIA COMPLETO*

Database file

Format for Screen

Query

Catalog

View

Quit dBASE III PLUS

A:

B:

C:

D:

E:

INCLUINDO:

* Estrutura completa dos Menus
(c/ tradução p/ o português).

Aplicações c/ MS-DOS.

SuperMacro Central .prg
(complexo de 12 programas .PRG).

Interações c/ WordStar 4,
Lotus 1-2-3 v. 2, dBASE II,
Sideways, SideKick, Turbo Basic
e Clipper.

Apresentação simultânea
(Menus X Comandos do *prompt*).

Programação consistente.

Fartamente ilustrado!

DISQUETE OPCIONAL

Consulte sua livraria mais próxima para
adquirir o Disquete Opcional contendo
todos os programas do livro prontos pa-
ra utilização com seu dBASE III PLUS.

2ª Reimpressão

IBM Books

EBRAS Editora Brasileira

Berkeley
Roger Dalton

dBASE III PLUS

GUIA COMPLETO

--

--

Roger Dalton

dBASE III Plus Guia Completo

Tradução, Adaptação e Execução de Programas:

Ricardo Reinprecht
Editor de Informática

Revisão Geral:

Gisella Narcisi
Editora Assistente

EBRAS

Do original:
dBASE Plus® – A Complete Guide
Copyright© 1988 by Berkeley Publishers.
Copyright© 1988 by EBRAS Editora Brasileira Ltda.

Todos os direitos de tradução reservados pela EBRAS Editora Brasileira Ltda.

Nenhuma parte desta publicação poderá ser reproduzida ou transmitida por qualquer modo ou meio, seja este eletrônico, fotográfico, mecânico, ou outros, sem autorização prévia e escrita da EBRAS – Editora Brasileira Ltda.

Qualquer semelhança dos nomes próprios que fazem parte deste livro com qualquer pessoa é mera coincidência.

Editor: Ricardo Reinprecht
Editora Assistente: Gisella Narcisi
Área Comercial: Wilson Roberto Paulino
Consultor Técnico: Ricardo Freire de Carvalho

Capa:
Criação: Ricardo Reinprecht

Diagramação/Composição/Produção: *RioText*

Marcas registradas:
dBASE III Plus e dBASE II são marcas registradas da Ashton-Tate, Inc.; Lotus 1-2-3 é marca registrada da Lotus Development Corporation; SideKick é marca registrada da Borland Inc.; Sideways é marca registrada da Funk Software, Inc.; Turbo Basic é marca registrada da Borland International, Inc.; WordStar Professional Release 4 é marca registrada da MicroPro International Corporation.

1988
EBRAS – Editora Brasileira, Importação, Exportação e Comércio de Livros Ltda.
Rua Borges Lagoa, 1.044 – Cep 04038 – Vila Mariana – Fones: (011) 549-8633 e
(011) 549-8408 – Telex DRBN (011) 25966 – São Paulo – SP.

À minha família.

Sumário

Capítulo 1 – INTRODUÇÃO

Banco de Dados	2
A Estrutura do Banco de Dados	2
Como Definir a Estrutura	3

Capítulo 2 – MS-DOS, PC-DOS

Interações com o Sistema Operacional	6
Corrigindo Erros Durante a Digitação	7
Utilização Prática do DOS	8
Manipulando o DOS	8
Formatando seu Disquete Virgem	11
Renomeando seu Arquivo	14
Copiando Arquivos	15
Apagando um Arquivo	16

Capítulo 3 – COMO ENTRAR E SAIR DO dBASE III PLUS

Como Carregar e Sair do Sistema	18
Micros com Dois Acionadores de Disquete	18
Micros com Disco Rígido	19
Sistemas em Rede	20
Movimentação na Tela do Menu	20
Como Sair do Menu e do Programa	20
A Tela do Menu Principal	21
A Tela Help	21

Capítulo 4 – NOSSO PRIMEIRO PROGRAMA

A Criação do Arquivo	24
O Nome do Arquivo	24
As Teclas de Controle do Cursor e de E/S	25
A Capacidade do dBASE III Plus	27
O Menu de Estruturação	28
A Linha de Status	28
Abrindo Arquivos com o Menu	32
Saindo da Tela e Armazenando o Arquivo	34

VIII

Capítulo 5 – MANIPULAÇÃO E PESQUISA NO ARQUIVO

Procedimentos para Usar o Menu	38
Manipulando Nomes	38
Pesquisando com LIST FOR	40
Especificando a Exibição dos Campos	44
Combinando .AND., .OR. e \$	49
Usando LOCATE	49
Programando as Teclas de Função	52

Capítulo 6 – TÉCNICAS DE CLASSIFICAÇÃO DE ARQUIVOS

Classificando com SORT	56
Apagando os Dados (a partir do <i>prompt</i>)	61
Evitando a Perda Acidental	65
Apagando Todo o Banco de Dados	66
Classificando com INDEX	66
Mudando a Ordem de Apresentação dos Campos	69
Vantagens de INDEX Sobre SORT	70
Reindexando Arquivos	72
Limitações	74
Classificações Múltiplas	74
Busca Rápida de Registros com FIND e Display	77
Mais Velocidade de Busca	79

Capítulo 7 – EDIÇÃO DO BANCO DE DADOS: ATUALIZAÇÃO E ALTERAÇÃO

Inserindo Nôvos Campos	83
Inserção no Meio do Arquivo	83
Eliminando Campos	84
Usando EDIT	85
Usando BROWSE	89
Fechando o Banco de Dados	91
Capacidade de Arquivos Ativos – As Áreas de Trabalho	91
Movimento Horizontal da Tela	93
Movimento Vertical da Tela	93
Usando CHANGE e REPLACE: Edição Global	94
Apagando Arquivos via Menu	97

Capítulo 8 – USANDO ARQUIVOS QUERY – DEPURANDO SUA PESQUISA

Analisando o Novo Arquivo	103
Cancelando um Arquivo Query	107
O Uso de Parênteses	108
Testando Suas Habilidades	110
Resultado Final da Tela Query	111

Capítulo 9 – NÚMEROS, DATAS E VARIÁVEIS DE MEMÓRIA

Somando Campos	115
Usando .AND. e .OR.	117
Usando SQRT	122
Usando ABS	122
Usando ROUND	123
Usando LOG e EXP	124
Usando AVERAGE	124
Usando COUNT	125
Operações com Datas	126
Indexação com Datas	133
Usando TOTAL	136
As Variáveis Públicas e Privadas	137
Iniciação à Programação	138

Capítulo 10 – dBASE III PLUS RELACIONAL: CRIAÇÃO E MANIPULAÇÃO DE BANCOS DE DADOS MÚLTIPLOS

Usando APPEND FROM	145
Usando JOIN WITH	148
Usando SET RELATION TO	151
Usando VIEW	153

Capítulo 11 – PROGRAMAÇÃO COM O dBASE III PLUS: MACROS E DECISÃO AUTOMATIZADA (O “SuperMacro” auto@)

Introdução à Programação	158
O Primeiro Arquivo de Comandos	158
Separando os Espaços	161
O Menu de Programação de MODIFY COMMAND	163
Macros: Programando Etiquetas de Endereçamento	164
O Arquivo de Endereçamento	164
Gerando Suas Próprias Etiquetas	167
Programa ETIQUETA.PRG – Aplicação de Macros	167
Algumas Dicas Importantes	169
auto@ o “SuperMacro”: Controle Automático de Decisões	170
Usando @, SAY e GET	170
Programando auto@	171
Conectando REF e auto@	173
Editoração de Arquivos com o dBASE III Plus: Capacidade Relativa	174
A Capacidade do Campo Memo	176

Capítulo 12 – IMPRESSÃO DE RELATÓRIOS: SIMPLES E MÚLTIPLOS

As Outras Opções de Report	188
Produção Rápida de Etiquetas Via Menu	191

Impressão Condicional	196
Impressão Via Prompt	196
Impressão de Arquivos Múltiplos	197
A Opção Options	198
A Opção Groups	198
A Opção Columns	199
Imprimindo Listagens de Programas (.PRG)	199

Capítulo 13 – PROJETO E DESENVOLVIMENTO DE PROGRAMAS INCLUINDO TÉCNICAS DE DEPURAÇÃO

Programando o Sistema	205
Programando o Menu Central	205
Programando CASO1.PRG	208
Programando CASO2.PRG	209
Programando CASO2A.PRG: Corrigindo ETIQUETA.PRG	210
Programando CASO2B.PRG	211
Programando CASO3.PRG	211
Programando CASO4.PRG	212
Programando NUM1.PRG	213
•Programando NUM2.PRG e NUM3.PRG	214
Programando Del@.prg	215
Técnicas de Depuração	216
A Memória dos Comandos do Teclado	219
Mais Depuração com Comandos SET	221
Como Abreviar Comandos	222
Usando Substrings	223
Lista de Comandos SET	223
Aumentando a Velocidade com Procedures	228

Capítulo 14 – CRIAÇÃO DE TELAS PERSONALIZADAS

O Screen Painter	236
As Formas de Criação	236
Trabalhando com os Campos de Screen Painter	239
Salvando seu Arquivo Parcial	239
Preenchendo os Novos Campos	243
Especificando Condições Especiais	243
Salvando seu Trabalho (novamente)	245
Modificando sua Tela Personalizada	246
Operando com sua Tela Personalizada	247
Alterando o Tamanho dos Campos	247
Alteração Definitiva	247
Alteração Provisória	248
Acrescentando Mais Campos em sua Tela Personalizada	248
Apagando Campos	249

Permanentemente	249
Provisoriamente	249
Usando o Campo MEMO com Screen Painter	250
Criando uma Tela Especial de Auxílio	250
Mais Opções do Menu Screen Painter	252
Trabalhando com Screen Painter a Partir do Prompt	254

Capítulo 15 – CONVERSÕES E INTERFACES

Interface dBASE III Plus-SIDEWAYS	261
Interface dBASE III Plus-Sidekick	264
Interface dBASE III Plus-Turbo Basic	268
Interface dBASE III Plus-Lotus 1-2-3	270
Transferência Lotus-dBASE	273
Conversão dBASE III Plus-WordStar	276
Um Sistema de Endereçamento Completo com o Merge	278
Usando o Sistema Completo de Endereçamento	281
Endereçamento Direto no Envelope	281
Formatos Padrão para Transferências de Arquivo	283
Usando IMPORT e EXPORT	284
Conversão dBASE II-dBASE III Plus	284
Conversões em Lote	284

Capítulo 16 – CLIPPER – COMPILANDO SEM ERROS

Conhecimentos Básicos do Programador	288
Ferramentas Básicas para Compilação	288
Compilando em Disquetes Flexíveis	288
Os Acionadores de Disquetes	289
Central.Prg: Comandos Inválidos no CLIPPER	289
Uma Rotina Especial da Formatação de Tela	296
Compilando o Sistema	300
“Linkedição” via DOS	301

Capítulo 17 – LAN – REDES LOCAIS COM dBASE III PLUS

Introdução às Redes Locais	304
Qual o Perfil Ideal do Usuário?	304
As Necessidades Básicas	304
A Instalação dos Programas	305
Instalando o Administrador	305
Instalando o ACCESS	306
Tipos de Servidores de Arquivo	306
Comandos de Inicialização	307
Inicialização Via Comandos de Lote	307
Acessando um Arquivo	308

Programação de Segurança	309
Usando FLOCK/UNLOCK	309
Usando RLOCK: Uma rotina de espera	309
O Utilitário Protect: Codificação de Segurança	310
Acessando um Arquivo Privilegiado	312
O Arquivo Criptografado	313
As Opções de Saída e Salvamento	313
Comandos e Funções do Administrador	314
Software Opcional	314

APÊNDICE A – ÍNDICE DE COMANDOS E FUNÇÕES DO dBASE III PLUS

Comandos	318
Índice de Funções por Categoria	329
Caracteres	329
Conversões	330
Data e Hora	330
Entrada de Dados Via Teclado	331
Identificação	332
Matemática	333
Símbolos e Sinais	333

APÊNDICE B – EXTENSÕES DE NOMES DE ARQUIVO

Extensões do dBASE III Plus	336
Arquivos Especiais	337
Extensões Gerais – DOS e Outros	337
Extensões dos Principais Programas	338
Extensões de Interface entre Arquivos	339
Códigos de Conversões e Interfaces	339

APÊNDICE C – ESTRUTURA COMPLETA DOS MENUS

Set Up	342
Create	342
Update	342
Position	344
Retrieve	344
Organize	345
Modify	345
Tools	346

APÊNDICE D – GUIA DE SERVIÇOS E INFORMAÇÕES ÚTEIS

Disquete Opcional	348
Distribuidores Nacionais	348
Instruções para Uso do Disquete Opcional	350

Principais Distribuidores de Software	351
Equipamentos Utilizados	352
Endereços dos Fornecedores	353
Índice Analítico	355
Seleções da Biblioteca EBRAS	357

Capítulo

1

Introdução

BANCO DE DADOS

Um banco de dados nada mais é do que um sistema de armazenamento e manipulação de informações, um *gerenciador* de dados.

Essas informações, os dados, podem ser de diferentes significados e utilidades, por isso, quanto mais eficiente for o gerenciamento das informações, melhor será seu aproveitamento.

Assim como um arquivo de aço que armazena, em gavetas separadas, por pastas alfabetizadas e por assunto (por exemplo, cada gaveta um assunto e para cada assunto um número *X* de pastas com os nomes e endereços dos clientes), o dBASE III Plus tem uma capacidade poderosa – e rápida – de lidar com esses arquivos de forma muito mais eficiente: com ele podemos alterar seus dados com incomparável rapidez. Com o dBASE III Plus, podemos inverter a ordem alfabética de nosso cadastro, alterar seus títulos, criar relatórios, apagar total ou parcialmente um arquivo, criar arquivos, traçar os últimos comandos emitidos no programa, corrigir sem rasuras, emitir etiquetas de endereçamento, controlar estoques e uma série de outras atividades, *tudo em frações de segundos!*

A Estrutura do Banco de Dados

Assim como imaginamos nosso arquivo de aço, cheio de gavetas e tomando um enorme espaço físico, veremos agora sua estrutura para, depois, a compararmos com a estrutura de um banco de dados.

Um arquivo de aço geralmente é ineficiente para conter todas as informações comerciais de uma empresa. Isto faz com que mais arquivos de aço sejam acrescentados à companhia. Dentro desses arquivos encontramos as gavetas. Cada gaveta contém uma dezena ou centena de pastas que, por sua vez, contém fichas. Estas contêm as informações, separadas geralmente por datilografia, em um cadastro ordenado. Toda vez que desejamos ler as informações de um determinado arquivo ou ficha, temos de nos deslocar até o arquivo, abrir a gaveta, procurar em ordem alfabética e retirá-lo para posterior exame.

Caso queiramos transferir informações da ficha para outro documento qualquer, temos de escrever novamente, copiando do original. Se quisermos apagar ou alterar alguma coisa teremos de usar borracha ou jogar a ficha fora e datilografar novamente, num processo muito demorado.

Com o dBASE III Plus isso não acontece, pois sua estrutura flexível, aliada ao microcomputador, nos oferece um espaço de memória compatível com nossos interesses e que ocupa somente uma ou duas “gavetas” de espaço, mais a mesa do micro e da impressora (que pode ser a da própria secretária ou assistente).

Toda vez que precisamos trabalhar com as informações de nosso arquivo, tudo o que te-

mos a fazer é pressionar algumas teclas e as temos à nossa disposição na tela do vídeo. Tudo sempre armazenado em disquete.

Veja como é a estrutura de arquivo do dBASE III Plus na Figura 1.1.

NOME	: Zé da Silva
ENDEREÇO	: Rua da Liberdade, 1888
TEL	: 1234567
CIDADE	: São Paulo
ESTADO	: SP
CEP	: 04038

Figura 1.1: Estrutura de arquivo de dBASE III Plus.

Nessa estrutura temos a ficha que representa o *registro* (*record*) das informações sobre Zé da Silva. Dentro do registro temos 6 *campos* (*fields*) que são NOME, ENDEREÇO, TEL, CIDADE, ESTADO e CEP.

Como Definir a Estrutura

A melhor forma de definir a estrutura do seu banco de dados é utilizando campos separados e independentes de informação para que, ao tentar manipular um determinado campo, o usuário não tenha de tocar em outras informações às quais ele deseja manter intactas. Por exemplo, ao colocar o campo Endereço e o campo Tel juntos, num só campo, qualquer alteração nesse campo implicará na modificação dos dois itens, o que, nesse caso, resultaria em dados truncados ou imprecisos. (Veja o Capítulo 4 para mais detalhes sobre a definição da estrutura do dBASE III Plus.)

Então, para estruturar adequadamente seu banco de dados, deve-se seguir o esquema da Figura 1.2:

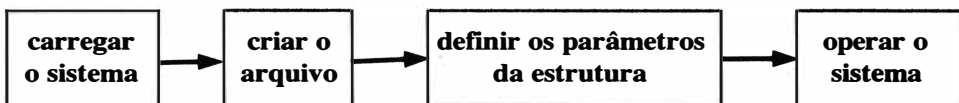


Figura 1.2: Esquema de estruturação de um arquivo.

onde **carregar o sistema** significa inserir (obviamente após o *prompt* do sistema operacional) o disquete do sistema 1 (*System 1*) do dBASE III Plus no acionador A (para outros acionadores – ou no caso de um só acionador –, consulte o Capítulo 3), após o *prompt* do dBASE para inserção do disquete do sistema 2 (*System 2*) inserir então o sistema 2 (é com este disquete que trabalharemos por todo o livro); **criar o arquivo** significa optar pela operação do dBASE III Plus via menu ou comandos do *prompt* e acionar o comando

CREATE para dar um nome a esse novo arquivo; **definir os parâmetros da estrutura** do arquivo significa estabelecer os tamanhos e as quantidades dos campos que comporão o arquivo, bem como seus tipos (veja mais detalhes no Capítulo 3 e 4); por último, **operar o sistema** significa conhecer a hierarquia das etapas necessárias para se chegar – o mais rápida e eficientemente possível –, ao resultado desejado.

São essas as etapas que tentaremos cobrir daqui por diante.

Capítulo

2

MS-DOS
PC-DOS

Interações com o Sistema Operacional

COPY	– Copia o arquivo
DIR	– Exibe o diretório (verticalmente)
DIR/W	– Exibe o diretório (horizontalmente)
DISKCOPY	– Copia e formata disquetes
ERASE	– Apaga arquivos
FORMAT	– Formata disquetes
FORMAT /V	– Formata e renomeia disquetes
RENAME	– Renomeia arquivos

Para utilizar seu dBASE III Plus, primeiro é necessário carregar seu sistema operacional MS-DOS, PC DOS, ou compatível. Isso pode ser conseguido a partir do disquete do sistema, ou *System Disk*, se preferir.

A seguir, com o computador desligado, insira o disquete do sistema no acionador A.

Após verificar se o disquete foi corretamente inserido e se a porta do acionador está fechada, ligue o computador e aguarde alguns segundos.

Após sua checagem interna, durante a qual a luz vermelha indicadora se acende e apaga confirmando a atividade, sua tela exibirá as seguintes mensagens:

```
XTEC SUPER TURBO V. 2.0
```

```
MS-DOS version 3.20
Copyright 1981,82,83,84,85 Microsoft Corp.
Current date is Tue 1-01-1980
Enter new date (mm-dd-yy):
```

que informa ao usuário a data atual *do sistema* e solicita a inserção de uma nova data, isto é, a data de *hoje*. Se quiser, insira a data *atual* após os dois pontos, onde se encontra o cursor.

O formato da data, mês-dia-ano, pode ser digitado de várias maneiras. Vamos supor que nossa data seja 9 de fevereiro de 1988. Você pode inserir na posição do cursor:

```
09-02-88
9-2-88
09/2/88
```

usando a barra (/) ou o traço (-) para separar os números.

Corrigindo Erros Durante a Digitação

Se você digitar algo errado, como uma data ou palavra indesejada, use a tecla BACKSPACE para retroceder e apagar os caracteres indevidos *antes* de pressionar ENTER. Caso você já tenha confirmado seus dados com ENTER, e não puder apagá-los com BACKSPACE, ou se o computador não tiver atendendo às solicitações do teclado, pressione ESC para cancelar a tela atual de informações e iniciar tudo outra vez. Como última alternativa, desligue o computador e inicie tudo novamente, ou pressione Ctrl-Alt-Del.

Após a data e o pressionamento da tecla ENTER (pode-se, também, ignorar a solicitação da data atual, indo direto à tela ENTER), sua tela exibirá a solicitação de preenchimento da hora atual:

```
XTEC SUPER TURBO V. 2.0
```

```
MS-DOS version 3.20
Copyright 1981,82,83,84,85 Microsoft Corp.
Current date is Tue 1-01-1990
Enter new date (mm-dd-yy):
Current time is 0:01:43.64
Enter new time:
```

Você pode simplesmente pressionar ENTER, ignorando a solicitação, ou preencher a hora atual do *seu* trabalho com o sistema da seguinte forma:

```
8:30:00.0
8:30:0
08:30:0
8:30
```

para 8:30 da manhã e

```
20:30:00.0
20:30:0
20:30
```

para a hora equivalente à noite.

Após o pressionamento de ENTER (com a inserção ou não da hora devida), sua tela deverá estar assim:

```
XTEC SUPER TURBO V. 2.0
```

```
MS-DOS version 3.20  
Copyright 1981,82,83,84,85 Microsoft Corp.  
Current date is Tue 1-01-1980  
Enter new date (mm-dd-yy):  
Current time is 0:01:43.64  
Enter new time:
```

```
Microsoft(R) MS-DOS(R) Version 3.20  
(C)Copyright Microsoft Corp 1981-1986
```

```
A>
```

Note que acima da mensagem do MS-DOS e sua versão utilizada neste exemplo (neste livro usamos também a versão 3.3 e o PC-DOS 3.3, alternadamente, nas diversas sessões de programação necessárias para autoria e testes), vemos a mensagem do computador sendo utilizado no momento da inicialização (no caso acima a mensagem “XTEC SUPER TURBO v. 2.0” indica o equipamento nacional utilizado para os testes da EBRAS – Editora Brasileira, dos programas do autor.

Após o *prompt* do sistema (A>), *caso tenha cópias de reserva de todos os programas a serem utilizados nesta sessão*, você pode retirar o disquete de seu sistema operacional (MS-DOS ou PC-DOS versões 1, 2 ou 3. Não há diferença quanto ao uso de qualquer dessas versões, já que só utilizaremos o sistema operacional para inicialização e algumas poucas operações básicas, neste livro). Após a retirada do disquete do sistema, caso você não queira fazer nenhuma cópia de programa, passe adiante para o próximo capítulo. Se precisar usar algumas capacidades do DOS, porém, consulte o restante deste capítulo.

UTILIZAÇÃO PRÁTICA DO DOS

Seu MS-DOS ou compatível tem uma utilidade *básica* que consideramos de fundamental importância para quem opera com aplicativos como planilhas eletrônicas, bancos de dados, processadores de texto, programas científicos etc. (ou seja, *tudo!*). A seguir relacionaremos algumas atividades consideradas pelo autor como *procedimentos mínimos*, que certamente facilitarão suas interações com o computador e seu aplicativo.

Manipulando o DOS

Primeiro, continuando a operação de inicialização do MS-DOS, vamos ver quais progra-

mas se encontram armazenados (gravados) no seu disquete. (Os programas podem variar de sistema para sistema. O nosso é só um exemplo.)

Para visualizar os programas do seu sistema operacional digite DIR. Sua tela deverá estar assim:

```
XTEC SUPER TURBO V. 2.0
```

```
MS-DOS version 3.20
Copyright 1981,82,83,84,85 Microsoft Corp.
Current date is Tue 1-01-1980
Enter new date (mm-dd-yy):
Current time is 0:00:53.93
Enter new time:
```

```
Microsoft(R) MS-DOS(R) Version 3.20
(C)Copyright Microsoft Corp 1981-1986
```

```
A>dir
```

Agora tecle ENTER e: 1) se você já tiver retirado o disquete do DOS do acionador de disquete (o *drive*), sua tela mostrará a mensagem abaixo; 2) se você ainda estiver com o sistema operacional no acionador A, sua tela exibirá o diretório de seu MS-DOS ou compatível como na figura seguinte. Por exemplo:

```
XTEC SUPER TURBO V. 2.0
```

```
MS-DOS version 3.20
Copyright 1981,82,83,84,85 Microsoft Corp.
Current date is Tue 1-01-1980
Enter new date (mm-dd-yy):
Current time is 0:00:53.93
Enter new time:
```

```
Microsoft(R) MS-DOS(R) Version 3.20
(C)Copyright Microsoft Corp 1981-1986
```

```
A>dir
```

```
Not ready error reading drive A
Abort, Retry, Ignore?
```

(Drive A não está pronto
Abortar, Repetir, Ignorar?)

Coloque o disquete do DOS novamente no acionador A e tecle “R” para que a operação seja repetida. Sua tela deverá exibir:

```
FORMAT  EXE    10973  1-01-80  12:04a
FIND     EXE     6403  3-21-86  12:00p
FC       EXE    14446  3-21-86  12:00p
GRAFTABL EXE     8210  3-21-86  12:00p
GRAPHICS EXE    13170  3-21-86  12:00p
JOIN     EXE     8942  3-21-86  12:00p
LINK     EXE    39172  3-21-86  12:00p
LABEL    EXE     2750  3-21-86  12:00p
MORE     COM      266  3-21-86  12:00p
MODE     EXE    13652  3-21-86  12:00p
PRINT    EXE     8824  3-21-86  12:00p
RAMDISK  SYS     1030  3-04-86  12:39a
RECOVER  EXE     4145  3-21-86  12:00p
REPLACE  EXE     4852  3-21-86  12:00p
SYS       COM     4607  1-01-80  12:31a
SHARE    EXE     8544  3-21-86  12:00p
SUBST    EXE     9898  3-21-86  12:00p
SORT     EXE     1898  3-21-86  12:00p
TREE     EXE     8556  3-21-86  12:00p
XCOPY    EXE     5396  3-21-86  12:00p
RELOGIO  COM     1378  11-13-85  12:29p
CONFIG   SYS      128  3-22-88   3:24p
      36 File(s)      20480 bytes free
```

A>

Observe que o comando DIR nos traz um diretório com 36 arquivos que não cabem, inteiramente, na tela. O que vemos são somente 22 arquivos. Para visualizar *todos* os arquivos de seu diretório simplesmente mude o formato da apresentação de seu diretório. Digite

A>dir/w

e sua tela ficará assim:

```
A>dir/w
```

```
Volume in drive A is DOS_3-2
Directory of A:\
```

```
COMMAND  COM   ANSI   SYS   ASSIGN  COM   ATTRIB  EXE   APPEND  COM
CHKDSK   EXE   CONFIG BAK   DEBUG   EXE   DRIVER  SYS   DISKCOMP EXE
DISKCOPY EXE   EDLIN   EXE   EXE2BIN EXE   FDISK   EXE   FORMAT  EXE
FIND     EXE   FC       EXE   GRAFTABL EXE  GRAPHICS EXE  JOIN    EXE
LINK     EXE   LABEL   EXE   MORE    COM   MODE    EXE   PRINT   EXE
RANDISK  SYS   RECOVER EXE   REPLACE EXE   SYS     COM   SHARE   EXE
SUBST    EXE   SORT    EXE   TREE    EXE   XCOPY   EXE   RELOGIO COM
CONFIG   SYS

      36 File(s)      20480 bytes free
```

```
A>
```

Agora você pode apreciar todos os 36 arquivos de seu MS-DOS.

Formatando seu Disquete Virgem

Para usarmos um disquete novo, o que o aconselhamos a fazer – para obter o maior espaço disponível possível para seus arquivos –, temos de *formatar* o disquete. Formatar significa preparar o disquete eletronicamente para receber os dados no seu tipo de computador. Só após a formatação é que o disquete estará pronto para operações de leitura/gravação.

Há duas maneiras de se formatar um disquete. Pode-se usar os comandos FORMAT ou DISKCOPY. Com o comando FORMAT, digitado a partir do *prompt*, formata-se um disquete no acionador A ou B (A>FORMAT ou A>FORMAT B:). Mas é com sua variação, FORMAT d:/V, que obtemos o melhor resultado. Com este comando podemos não só formatar o disquete como também dar-lhe um nome, conhecido como nome de volume, ou seja o nome, gravado eletronicamente, do disquete que conterà seus futuros arquivos. Por exemplo, para formatar seu disquete virgem no acionador B, com o DOS no acionador A, digite:

```
A>format b:/v
```

Sua tela deverá estar assim:

```
A>format b:/v
Insert new diskette for drive B:
and strike ENTER when ready
```

Pressione ENTER e o computador começará a formatar seu disquete. Repare que sua tela exibe mensagens sobre o processo de formatação como nos dois exemplos abaixo.

A>format b:/v	A>format b:/v
Insert new diskette for drive E:	Insert new diskette for drive E:
and strike ENTER when ready	and strike ENTER when ready
Head: 0 Cylinder: 22	Head: 1 Cylinder: 30

Ao terminarmos a formatação, seu computador exibirá a seguinte tela:

```
A>format b:/v
Insert new diskette for drive E:
and strike ENTER when ready

Format complete

Volume label (11 characters, ENTER for none)?
```

Digite o nome de seu disquete. Sugerimos DALTONBK1DB (arquivo geral **Dalton, Backup**, número 1, do programa **dBase**). Use no máximo 11 caracteres, como sugere a mensagem do DOS. Pressione ENTER.

Sua tela deverá estar assim:

```
A>format b:/v
Insert new diskette for drive E:
and strike ENTER when ready

Format complete

Volume label (11 characters, ENTER for none)? DALTONBK1DB

362496 bytes total disk space
362496 bytes available on disk

Format another (Y/N)?
```

Responda “N” à pergunta sobre se você deseja formatar outro disquete. No caso de querer formatar tecle “Y” e repita simplesmente a operação de nomeação do disquete dando-lhe o nome, por exemplo, de DALTONBK2DB (ou qualquer outro nome que lhe seja mais sugestivo).

O comando DISKCOPY formata o arquivo enquanto copia *integralmente* o disco fonte. Ele não permite que lhe demos um nome de volume porque seu nome de volume será, obviamente, o próprio nome do disquete sendo copiado. É o comando mais indicado para cópias de discos inteiros como no caso da cópia reserva de seu MS-DOS (caso ainda não a tenha feito, use DISKCOPY agora e faça uma cópia reserva de seu sistema operacional.

Não fique à disposição de quedas repentinas de energia ou outros “bichos” que podem acabar com sua alegria e com seu sistema!)

Aplique DISKCOPY agora. Digite

```
A>diskcopy a:b:
```

e pressione ENTER. Note a mensagem abaixo:

```
A>diskcopy a:b:
```

```
Invalid parameter
```

```
Do not specify filename(s)
```

```
Command Format: DISKCOPY d: d: [/1]
```

```
A>
```

Você tem de dar espaços entre as letras dos acionadores (A: B:) ou seu sistema operacional não compreenderá a operação corretamente.

Digite novamente e a seguir pressione ENTER. Sua tela deverá exibir as seguintes mensagens:

```
A>diskcopy a: b:
```

```
Insert SOURCE diskette in drive A:
```

```
Insert TARGET diskette in drive B:
```

```
Press any key when ready . . .
```

O sistema operacional está solicitando a você que insira o disquete FONTE (o original) no acionador A e o disquete ALVO (o reserva) no acionador B. Após cumprir com todos os procedimentos, pressione ENTER. Note que uma mensagem específica diz que seu sistema está copiando 40 trilhas, 9 setores por trilha, nos dois lados:

```
A>diskcopy a: b:
```

```
Insert SOURCE diskette in drive A:
```

```
Insert TARGET diskette in drive B:
```

```
Press any key when ready . . .
```

```
Copying 40 tracks
```

```
9 Sectors/Track, 2 Side(s)
```

Ao término da sessão, uma pergunta:

```
Copy another diskette (Y/N)?
```

(Copia outro disquete (S/N?))

responda com “N” ou “S” caso você queira ou não copiar outro disquete importante.

Renomeando seu Arquivo

Um outro comando importante é RENAME, que nos permite renomear um arquivo. Por exemplo nosso diretório do DOS contém um programa denominado RELÓGIO.COM. Vamos renomeá-lo para CLOCK.COM. Digite RENAME RELÓGIO.COM CLOCK.COM. Caso seu disquete do DOS esteja com a pequena etiqueta de proteção contra gravação, retire-a para que o disquete aceite essa operação de gravação. Nesse caso sua tela ficará assim:

```
A>RENAME RELOGIO.COM CLOCK.COM
```

```
Write protect error writing drive A
Abort, Retry, Ignore?
```

Após retirar a proteção, tecle R. Sua tela deverá estar assim:

```
A>RENAME RELOGIO.COM CLOCK.COM
```

```
Write protect error writing drive A
Abort, Retry, Ignore? R
```

```
A>
```

e seu programa RELÓGIO.COM já terá mudado de nome. Digite DIR e veja o penúltimo arquivo de seu DOS.

```

FORMAT  EXE    10973   1-01-80  12:04a
FIND     EXE     6403   3-21-86  12:00p
FC       EXE   14446   3-21-86  12:00p
GRAFTABL EXE     8210   3-21-86  12:00p
GRAPHICS EXE   13170   3-21-86  12:00p
JOIN     EXE     8942   3-21-86  12:00p
LINK     EXE   39172   3-21-86  12:00p
LABEL    EXE     2750   3-21-86  12:00p
MORE     COM      266   3-21-86  12:00p
MODE     EXE   13652   3-21-86  12:00p
PRINT    EXE     8824   3-21-86  12:00p
RAMDISK  SYS     1030   3-04-86  12:39a

```

```

RECOVER  EXE      4145   3-21-86  12:00p
REPLACE  EXE      4852   3-21-86  12:00p
SYS       COM      4607   1-01-80  12:31a
SHARE    EXE      8544   3-21-86  12:00p
SUBST    EXE      9898   3-21-86  12:00p
SORT     EXE      1898   3-21-86  12:00p
TREE     EXE      8556   3-21-86  12:00p
XCOPY    EXE      5396   3-21-86  12:00p
CLOCK    COM      1378   11-13-85  12:29p
CONFIG   SYS       128    3-22-88   3:24p
        36 File(s)    20480 bytes free

```

A>

Copiando arquivos

Uma prática muito comum é copiar um arquivo de um disquete para outro. Fazemos isso com o comando COPY. A instrução

```
A>COPY A:CLOCK.COM B:
```

nos daria a seguinte tela:

```

A>COPY A:CLOCK.COM B:
      1 File(s) copied

```

A>

confirmando que 1 arquivo foi copiado.

Para verificar se realmente CLOCK.COM foi copiado de nosso sistema operacional no *drive* A para nosso disquete de trabalho DALTONBK1DB no acionador B (ou *drive* B, tanto faz!), digite DIR B: e visualize a tela:

```

A>COPY A:CLOCK.COM B:
      1 File(s) copied

A>DIR B:

Volume in drive B is DALTONBK1DB
Directory of B:\

CLOCK   COM      1378   11-13-85  12:29p
      1 File(s)    360448 bytes free

```

A>

Apagando um arquivo

No exemplo anterior aprendemos a copiar arquivos e copiamos, como exercício, o arquivo CLOCK.COM no disquete de trabalho DALTONBK1DB. Na realidade não precisaremos desse arquivo em nossas interações com o dBASE III Plus, portanto, vamos apagá-lo! Acione o comando

```
A>ERASE B:CLOCK.COM
```

e a seguir digite DIR B: para visualizar o resultado.

```
A>DIR B:
```

```
Volume in drive B is DALTONBK1DB  
Directory of B:\
```

```
File not found
```

```
A>
```

A última mensagem diz que o arquivo não foi mais encontrado.

Agora, esperamos que você esteja pronto para iniciar-se no mundo fascinante do banco de dados mais popular do mundo, o dBASE III Plus. Boa sorte.

Capítulo

3

**Como Entrar e Sair
do dBASE III Plus**

COMO CARREGAR E SAIR DO SISTEMA

Leia a parte referente à Instalação e Configuração (CONFIG.SYS) do dBASE III Plus em seu hardware. Os procedimentos são especificados para micros com acionadores de disquetes, discos rígidos ou sistemas ligados em rede.

Micros com Dois Acionadores de Disquetes

Insira o disquete do Sistema 1 no acionador A e um disquete em branco, formatado, no acionador B. Após o *prompt* A> do MS-DOS ou do PC-DOS, digite

DBASE ENTER

e a seguinte “tela” aparecerá em seu vídeo:

dBASE III PLUS version 1.0 IBM/MSDOS
 Copyright (c) Ashton-Tate 1984, 1985, 1986. All Rights Reserved.
 dBASE, dBASE III, dBASE III PLUS, and Ashton-Tate
 are trademarks of Ashton-Tate

You may use the dBASE III PLUS software and printed materials in the dBASE III PLUS software package under the terms of the dBASE III PLUS Software License Agreement. In summary, Ashton-Tate grants you a paid-up, non-transferable, personal license to use dBASE III PLUS on one microcomputer or workstation. You do not become the owner of the package, nor do you have the right to copy or alter the software or printed materials. You are legally accountable for any violation of the License Agreement or of copyright, trademark, or trade secret laws.

Press the F1 key for HELP.
 Insert System Disk 2 and press ENTER, or press Ctrl-C to abort

Abaixo da nota de *Copyright*, a mensagem em seu vídeo diz:

Insert System Disk 2 and press ENTER, or press Ctrl-C to abort
 (insira o disquete do Sistema 2 e pressione ENTER, ou pressione Ctrl-C para cancelar a operação)

A seguir remova o Sistema 1 do acionador A e coloque o disquete do Sistema 2, de acordo com a mensagem, e pressione ENTER. A tela do menu principal aparecerá com 8 opções principais no topo, a serem escolhidas pelo usuário. (caso você se encontre no *prompt* (“.”) do dBASE, digite ASSIST.) Agora veja a tela do menu principal do dBASE III Plus:

Set Up Create Update Position Retrieve Organize Modify Tools 01:03:41 am

Database file
Format for Screen Query
Catalog View
Quit dBASE III Plus

ASSIST	:	<A:>	:		:	Opt: 1/6	:		:
--------	---	------	---	--	---	----------	---	--	---

Move selection bar - ↑↓ . Select - ← . Leave menu - → . Help - F1. Exit - Esc.
Select a database file.

Experimente as teclas de movimentação do cursor (a tecla com a seta à direita) e pressione-a algumas vezes. A cada pressionamento um dos comandos principais é destacado (iluminado).

Agora continuaremos nossa instrução de inicialização. Mais adiante retornaremos à tela do dBASE III Plus para construirmos nosso primeiro programa.

Micros com Disco Rígido

Para micros de disco rígido, utilize os comandos CHDIR ou CD, do seu DOS, e ative o diretório apropriado do seu disco rígido.

O *prompt*

C>

aparecerá na tela. Digite o comando

DBASE ENTER

Após a mensagem de *copyright* pressione ENTER para visualizar o menu principal do dBASE, o ASSIST.

Sistemas em Rede

Procure em seu manual original do dBASE III Plus, as informações necessárias para a inicialização do seu programa em um sistema integrado por redes de micros.

Agora, retornaremos às 8 opções do menu principal.

Set Up Create Update Position Retrieve Organize Modify Tools

Movimentação na Tela do Menu

Há duas formas iniciais de movimentar o cursor na tela do menu do dBASE III Plus. Pode-se usar as teclas com as setas para a direita e para a esquerda, ou teclar a primeira letra do comando. Experimente.

Tecla C

O comando *Create* é iluminado.

Tecla U

O comando *Update* é iluminado.

Tecla M

O comando *Modify* é iluminado, etc.

Da mesma forma que movemos o cursor para a direita ou esquerda, podemos movê-lo para cima e para baixo, após a seleção de uma das opções. Lembre-se, qualquer opção para ser selecionada precisará do pressionamento da tecla ENTER (ou RETURN) como ação final.

Como Sair do Menu e do Programa

Para sair do menu principal pressione a tecla Esc. O menu desaparecerá e o *prompt* aparecerá em seu lugar. (Qualquer comando pode ser digitado a partir do *prompt* “. ”). Digite ASSIST ENTER novamente para retornar ao menu principal. Para sair do programa e voltar ao *prompt* do DOS, escolha a opção *Set Up* do menu principal e selecione *Quit dBASE III Plus* ou simplesmente digite o comando

QUIT ENTER

Caso você esteja no *prompt* “. ”, digite ASSIST primeiro e depois proceda a operação com QUIT.

A Tela do Menu Principal

A tela do menu ASSIST do dBASE III Plus é composta por:

- 8 opções principais
- Subopções (para cada opção principal)
- Linha de *status* com informações sobre o estado atual das operações e comandos. Por exemplo, qual das 6 subopções de *Set Up*, está sendo destacada (se for *Query* a linha de *status* indicará “Opt: 3/6 – a terceira opção das seis disponíveis, etc.”).

A Tela Help

A tela Help pode ser ativada a qualquer momento da escolha de uma opção do menu. Basta pressionar a tecla F1 após destacar a opção desejada. Experimente, destaque uma das opções do menu principal usando a primeira letra da opção ou a tecla com a seta. A seguir pressione F1 e veja as informações sobre o comando escolhido. Para continuar com a tela do menu anterior, pressione qualquer tecla – como diz a informação abaixo da linha de *status*. Assim, se você precisar de um lembrete rápido, não será necessário usar o manual, é só teclar F1.

Bem, essas foram informações *elementares* que tentamos passar, nesses três primeiros capítulos. Agora, esperamos que você já esteja pronto para, a partir do Capítulo 4, iniciar suas interações com o dBASE III Plus. Não se esqueça de consultar as seções anteriores sempre que for necessário.

Capítulo

4

Nosso Primeiro Programa

Para criar nosso primeiro banco de dados, siga os seguintes procedimentos:

Primeiro insira o disquete do Sistema 1 do dBASE III Plus, após o *prompt* A> ou B> do seu acionador de disquete. Digite DBASE (ou dbase, tanto faz!) e aguarde até que ele seja carregado. Após a mensagem de *Copyright* da Ashton-Tate, à solicitação “*Insert System 2 ...*”, troque o Sistema 1 pelo Sistema 2 e pressione ENTER. É com este disquete que você trabalhará a maior parte do tempo.

Neste momento o dBASE já deverá estar com sua tela do menu principal, o ASSIST, em ação. (Caso não tenha tido sucesso, verifique e repita todos os procedimentos novamente...).

Destaque a opção *Create*, iluminando-a com o cursor (tecla com a seta) ou simplesmente teclando C.

Create

ou

C

Selecione a opção *Database file* (arquivo de banco de dados) pressionando ENTER (a opção desejada deve estar em destaque – use as teclas com as setas verticais).

Agora a sua tela deverá estar exibindo as opções A:, B: e C:. Selecione B: (para micros com dois acionadores) ou C: (para disco rígido).

A CRIAÇÃO DO ARQUIVO

O Nome do Arquivo

Agora a mensagem

Enter the name of the file

está aguardando a digitação do nome que você deseja dar ao arquivo. Utilize *qualquer* palavra com até *oito* caracteres. Não use espaços em branco ou símbolos de pontuação.

Em nosso exemplo utilizaremos o arquivo CLIENTES. Digite

CLUEMTEs ENTER

ATENÇÃO: caso você também cometa este erro, *antes* de teclar ENTER use a tecla BACKSPACE de seu micro para apagar o nome errado. Pressione BACKSPACE até a le-

tra U (inclusive). A seguir digite o nome correto e pressione ENTER.

CL
CLIENTES ENTER

As Teclas de Controle do Cursor e de E/S

Vimos no capítulo anterior a aplicação da tecla ESC e do comando QUIT para sair do sistema, parcial ou totalmente. Para movimentar o cursor para cima, para baixo, ou para os lados use as teclas com as setas, que fazem parte do teclado numérico de seu PC, geralmente à direita do teclado principal. BACKSPACE é usada para apagar os caracteres digitados errados.



Figura 4.1: Exemplo de Teclado.

A seguir relacionamos, na Tabela 4.1, as teclas de controle e de E/S. Os comandos entre parênteses representam as teclas alternativas que produzem o mesmo efeito das teclas da primeira coluna. O símbolo “^” representa a tecla Ctrl (control) e quer dizer “pressione Ctrl e, enquanto a mantém pressionada, pressione a tecla com a letra correspondente ...”.

← (parte superior do teclado) ou BACKSPACE		Move o cursor para trás apagando os caracteres.
↑	(^E)	Move o cursor para cima.
↓	(^X)	Move o cursor para baixo.
←	(^S)	Move o cursor para a esquerda.
→	(^D)	Move o cursor para a direita.
Del	(^G)	Deleta o caracter sob o cursor.
Ins	(^V)	Ativa/desativa o modo "Insert".
PgUp	(^K)	Move-se para o registro anterior.
PgDn	(^C)	Move-se para o registro seguinte.
^End	(^W)	Salva todos os dados e retorna ao menu.
Home		Move-se para a primeira letra da palavra.
End		Move-se para a última letra da palavra.
^Y		Deleta os caracteres à direita.
^U		Deleta todo o registro.

Tabela 4.1: As teclas de controle do cursor e de E/S.

Como Preencher o Arquivo

Assim que você recebeu a mensagem *"Enter the name of the file"* e após digitar CLIENTES ENTER, o dBASE exibiu sua tela com o Formulário de preenchimento da estrutura do seu banco de dados.

Cada estrutura de um programa é definida pelo próprio usuário. Caberá a ele definir o *tamanho*, a *quantidade* e o *tipo* de informação que ele necessitará dentro do computador, incluindo quantas linhas de informação (*campos*) e colunas (*registros*).

Agora vamos trabalhar na estrutura de nosso programa e a seguir, analisar a tela de informações que temos em nosso vídeo. Iremos operar com esta parte da tela:

Field Name	Type	Width	Dec
1			

Isto significa que precisamos definir os nomes dos campos (*Field Name*), o tipo da informação (*Type*), o tamanho (*Width*) e a quantidade de casas decimais (*Dec*), se houver.

Nosso primeiro registro conterá os dados sobre "Zé da Silva", nosso cliente hipotético. Usaremos 10 campos para definir este arquivo: Profissão, Sexo, Nome, Sobrenome, Tel., Endereço, Bairro, Cidade, Estado e CEP: (Você só pode usar dez caracteres de comprimento, letras ou números, para cada nome de campo. Nenhum símbolo de pontuação pode ser usado, exceto o traço "-" e o nome do campo deve começar com uma letra).

Pode haver cinco tipos diferentes: Caracteres (*Character*) – usado para informações não numéricas como os nomes de campo; Numérica (*Numeric*) – usado para números aos quais aplicaremos operações matemáticas, como valores em dinheiro, controle de estoque, etc.;

Data (*Date*) – usado para datas no formato MM/DD/YY (dois caracteres para o mês, dois para o dia e dois para o ano); *Memo* – para textos extensos somente; Lógico (*Logical*) – para operações condicionais do tipo verdadeiro ou falso.

Em nosso caso, usaremos o primeiro tipo, caracteres, porque não faremos nenhuma operação lógica ou matemática com os dados que preencheremos e nem usaremos nenhuma data ou formato extenso de texto. Portanto, preencha os tipos e seu tamanho, *width*, como segue:

Field Name	Type	Width	Dec	Field Name	Type	Width	Dec
1	PROFISSAO	Character	25	9	ESTADO	Character	2
2	SEXO	Character	10	10	CEP	Character	6
3	NOME	Character	15				
4	SNOME	Character	15				
5	TEL	Character	11				
6	ENDereco	Character	25				
7	BAIRRO	Character	15				
8	CIDADE	Character	15				

Como isto é só o que queremos no momento, pressione ENTER duas vezes. A mensagem

Press ENTER to confirm. Any other key to resume

(Pressione ENTER para confirmar. Qualquer outra tecla para finalizar)

aparecerá na sua tela. Pressione ENTER. A mensagem

Input data records now? (Y/N)

(Inserir os registros agora? (Sim/Não))

Você tem duas chances. Se pressionar Y (*Yes*) seu primeiro arquivo estará aberto para que você possa preenchê-lo. Teclando N (*Não*), você retornará para o menu principal e todas as informações que acabou de digitar ficarão guardadas até que decida *abrir* seu banco de dados.

Tecla “N” e retorne para o menu principal.

Antes de prosseguirmos com nosso recém-criado banco de dados, vamos analisar os dados importantes de nossa tela durante nossa estruturação do programa CLIENTES.

A Capacidade do dBASE III Plus

Antes de nossa operação de preenchimento dos campos e definição da estrutura, o canto direito superior da tela exibia a seguinte mensagem: “*Bytes remaining: 4000*”. Isto significa que, no início, tínhamos 4000 caracteres disponíveis para o preenchimento de nossos campos.

O dBASE III Plus pode armazenar até 128 campos e um total de 4000 caracteres em cada registro. Eram a esses 4.000 caracteres que a mensagem se referia.

Agora, ao visualizar a estrutura de seu arquivo CLIENTES novamente, você pode ver que a mensagem diz que há somente 3861 bytes disponíveis. Isto significa que você já consumiu 139 bytes ou caracteres quando definiu a estrutura de seu primeiro arquivo. Seu registro terminará quando você utilizar os 128 campos (linhas) disponíveis, ou alcançar os 4000 caracteres – o que acontecer primeiro. Se quiser verificar como você fez isso, é só somar os números da coluna *width* da estrutura que acabamos de criar.

O Menu de Estruturação

O menu de estruturação do dBASE III Plus traz, acima da estrutura, um retângulo com quatro quadros contendo opções de controle do cursor, inserção, deleção e movimentação de campos, saída e cancelamento. Vejamos seu significado na Tabela 4.2.

CURSOR	
Char:	Movimentação de um espaço à esquerda e à direita.
Word: Home End	Movimentação para o início da palavra (Home) e para o fim da palavra (End).
Pan: ^← ^→	Obtido com “Ctrl+seta”. Permite visualizar um campo oculto pelas limitações físicas do vídeo.
INSERT	
Char: Ins	Ativa/desativa o modo de inserção. Pode ser usado como Ctrl-V.
Field: ^N	Ctrl-N. Insere um campo novo na posição do cursor.
Help: F1	Chama a tela de auxílio sobre o comando sendo utilizado.
DELETE	
Char: Del	Del Ctrl-G. Deleta o caractere sobre o cursor.
Word: ^Y	Deleta a palavra à direita do cursor.
Field: ^U	Deleta o campo correspondente ao cursor.
Up a field:	Move para cima um campo.
Down a field:	Move para baixo um campo.
Exit/Save: ^End	Ctrl-End ou Ctrl-W. Sai do programa e mantém o trabalho já feito.
Abort: Esc	Também Ctrl-Q. Sai do programa sem gravar os dados.

Tabela 4.2: Descrição dos comandos do menu de estruturação.

A Linha de Status

Enquanto trabalhamos com nossa tela, durante o processo de estruturação, a *linha de status*, a linha reta que contém algumas informações sobre nossa tela atual, disposta na parte

inferior do vídeo, mostrava as seguintes informações, no final do preenchimento do décimo campo:

CREATE	<B:>	CLIENTES	FIELD: 10/10
--------	------	----------	--------------

CREATE: Nome do comando atual.
 B: Acionador B ativo.
 CLIENTES: Nome do arquivo.
 Field: 10/10 Posição do cursor, décimo campo de um total de dez.

Abaixo da linha de *status* vimos a mensagem

Enter the field name

que diz para o usuário preencher os nomes dos campos.

A última mensagem,

**Field names begin with a letter and may
 contain letters, digits and underscores**

comunica ao usuário que os nomes de campo devem iniciar com uma letra e podem conter letras, dígitos e traços somente.

Aí está, uma análise completa de nossa tela!

Agora retornemos ao nosso programa. Vamos aprender a usar nosso programa diretamente, isto é, emitindo comandos a partir do *prompt* (.) sem usar o menu principal.

Primeiro, ainda no menu principal, tecele ESC. Sua tela atual desaparecerá e em seu lugar surgirá uma tela vazia com o *prompt* “ . ”. (Não é bem uma tela vazia, apenas o menu desaparece, permanecendo a área informativa da parte inferior da tela.)

Vamos abrir nosso banco de dados para podermos inserir os dados com os quais trabalharemos nesta fase. Caso esteja operando com um sistema de disco rígido, digite o comando.

USE CLIENTES ENTER

ou, se o seu sistema for de disquetes, especifique o acionador B (caso você ainda não o tenha especificado):

SET DEFAULT TO B ENTER

para dizer ao programa que seu disquete está no acionador B. A seguir acione o comando

USE CLIENTES ENTER

Neste momento já estamos prontos para começar a preencher os campos definidos em nossa estrutura.

Agora vamos utilizar o comando APPEND para recebermos autorização do programa para começar a preenchê-lo. Digite

APPEND ENTER

e você receberá um formulário com seu registro em branco, com as especificações que criamos no início.

Apesar de estarmos no modo do *prompt*, a execução de APPEND nos coloca na mesma tela de Append do menu Update¹. Isto acontece porque o dBASE não poderia, por simples bom senso, separar uma área de memória para cada comando, paralelamente. Compare com os esquemas das Figuras 4.2 e 4.3.

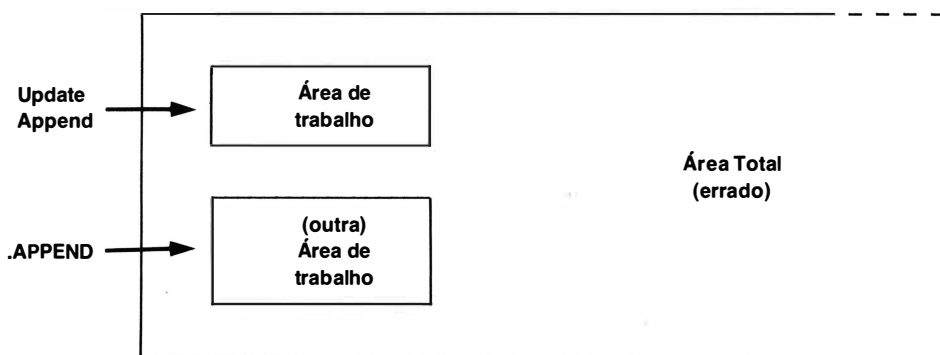


Figura 4.2: Exemplo figurativo de duas áreas de trabalho separadas para um mesmo comando.

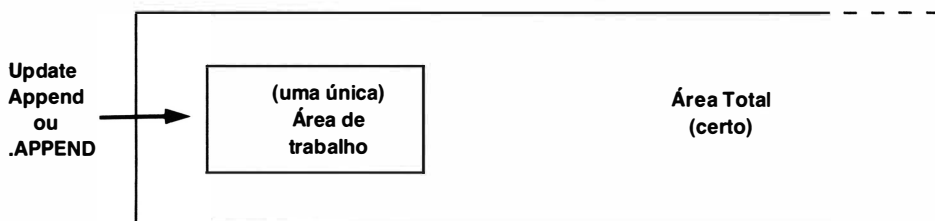


Figura 4.3: Exemplo figurativo de uma única área de trabalho alocada para os dois modos de acesso.

N.A. No Capítulo 6 falaremos mais sobre APPEND, com aplicação do MENU.

¹ Excetuando-se, obviamente, o espaço de apresentação (i.e., o *desenho* da tela) alocado para cada modo.

Por esta razão e para facilitar a compreensão pelo usuário, a maioria de nossas ilustrações, como nos exemplos de preenchimento dos registros no modo APPEND, aparecerão na sua posição correta, porém, sem os destaques das diferenças de tonalidade de luz proporcionadas pela posição atual do cursor em uma determinada opção.

Vamos preencher os campos com as seguintes informações:

PROFISSAO	Engenheiro	ENTER
SEXO	Masculino	ENTER
NOME	Jose	ENTER
SOBRENOME	Silva	ENTER
TEL	1234567	ENTER
ENDERECO	Rua da Calamidade, 13	ENTER
BAIRRO	Barbaridade	ENTER
CIDADE	Ficticia	ENTER
ESTADO	IM	ENTER
CEP	04001	ENTER

Agora o que você tem é um outro registro (*Record No. 2*) em branco para preencher seus próximos campos. Não importa quantos registros couberem no disquete, todos pertencerão ao mesmo arquivo (CLIENTES) até que criemos outro arquivo.

No momento temos o arquivo CLIENTES.DBF (falaremos sobre a *extensão* DBF e outras mais adiante) que é composto, por enquanto, de um registro com 10 campos com 138 bytes de uso.

Vamos digitar nossa segunda “ficha”. Siga o mesmo procedimento do registro 1 e insira:

Aviador
 Masculino
 Celso
 DeLanteuill
 2737747
 Rua das Águias, 767
 J. Botânico
 Rio de Janeiro
 RJ
 20000

Para testar nossos conhecimentos de controle do cursor, vamos retornar ao registro 2 (caso você já tenha pressionado ENTER após o CEP) e corrigir o sobrenome desse nosso cliente, pois ele foi digitado com erro: Pressione a tecla PgUp e retornaremos ao registro anterior (qualquer dúvida quanto às teclas de controle, veja a relação de comandos de controle e de E/S, na Tabela 4.1).

PgUp

Pronto! Aí está nosso registro 2 novamente. Agora, usando as teclas de controle (as setas, Ctrl-E – Ctrl-X, dependendo da posição de seu cursor), vá até o SOBRENOME de nosso cliente: se o cursor estiver no alto, em PROFISSÃO, pressione a seta para baixo (ou Ctrl-X) três vezes. Agora posicione o cursor para o final do sobrenome pressionando a tecla End uma vez. Se o sobrenome de nosso amigo Celso fosse composto de duas palavras, teríamos de pressionar End duas vezes). Veja toda a sequência abaixo:

PROFISSAO	:Aviador	(↓ ou Ctrl-X três vezes)
SOBRENOME	:DeLanteuill	(End uma vez)
SOBRENOME	:DeLanteuill_	

Agora vamos terminar a sequência, corrigindo o erro. Tecle ← duas vezes

SOBRENOME	:DeLanteuill	(← duas vezes)
-----------	--------------	----------------

e a seguir pressione Del ou Ctrl-G para que a letra l em excesso seja eliminada.

SOBRENOME	:DeLanteuil_
-----------	--------------

Seu Registro 2 deverá estar assim:

Record No.	2	
PROFISSAO	:Aviador	:
SEXO	:Masculino	:
NOME	:Celso	:
SOBRENOME	:DeLanteuil	:
TEL	:2737747	:
ENDereco	:Rua das Aguias, 767	:
BAIRRO	:J. Botanico	:
CIDADE	:Rio de Janeiro	:
ESTADO	:RJ:	
CEP	:20000:	

Para continuar o programa, após corrigir o SOBRENOME do Comandante Celso, pressione PgDn ou ENTER para ir para o Registro 3.

Abrindo Arquivos com o Menu

Para abrir seus arquivos com o menu principal, a partir da tela atual, digite END (Ctrl-END) ou ^W para salvar seu trabalho e retornar ao *prompt*. A seguir digite

ASSIST ENTER

Selecione a opção *Set Up* (com a seta ou teclando “S”) A seguir selecione a opção *Data-base file*. Sua tela exibirá três opções:

- A: (um acionador)
 B: (dois acionadores-acionador B)
 C: (disco rígido)

Após a seleção sua tela exibirá os arquivos armazenados no disquete atual. Selecione CLIENTES.DBF e pressione ENTER; A seguir responda N à pergunta

Is the file indexed? (Y/N)

porque não trabalharemos, por hora, com arquivos indexados.

CLIENTES.DBF está aberto e pronto para uso.

Já usamos a opção *Set Up* do menu principal para abrir CLIENTES.DBF; usamos *Create* para criar o arquivo; agora vamos usar a opção *Update* para selecionar o comando *Append* e retornar aos nossos registros. Destaque a opção *Update*. Do sub-menu de *Update* destaque a opção *Append*.

Agora você deverá visualizar sua tela do comando *Append*. Preencha o registro três. Aqui vai uma sugestão:

Record No.	3
PROFISSAO	:Editora :
SEXO	:Feminino :
NOME	:Gisella :
SOBRENOME	:Narcisi :
TEL	:5498633 :
ENDERECO	:Rua Borges Lagoa, 1044 :
BAIRRO	:Vila Mariana :
CIDADE	:Sao Paulo :
ESTADO	:SP:
CEP	:04038:

Ainda analisaremos mais três telas com seus respectivos comandos, para que você fique completamente familiarizado com seu significado. Vejamos agora o significado da tela de APPEND.

CURSOR

Char: ← → Ctrl-S e Ctrl-D. Retrocede ou avança um caractere.
Word: Home End Ctrl-F e Ctrl-A. Avança ou retrocede uma palavra.

UP DOWN

Field: Ctrl-E e Ctrl-X. Retrocede ou avança um campo.
Page: PgUp PgDn Ctrl-R e Ctrl-C. Retrocede ou avança um registro.
Help: F1 Ativa a tela Help.

DELETE	
Char: Del	Ctrl-G. Apaga o caractere sob o cursor.
Field: ^Y	Apaga o campo à direita do cursor.
Record: ^U	Apaga todo o Registro.
Insert Mode: Ins	Ctrl-V. Ativa/desativa o modo Insert.
Exit/Save: ^End	Ctrl-W. Sai da tela e armazena o Registro. Se um campo tiver sido preenchido, move-se para o próximo Registro.
Abort: Esc	Ctrl-Q. Sai da tela sem acrescentar o Registro.
Memo: ^Home	Chama o modo Memo.

APPEND	<B:>	CLIENTES	Rec: None
--------	------	----------	-----------

APPEND	Nome do comando atual
B:	Acionador em uso.
CLIENTES	Nome do arquivo.
Rec: None	Registro sendo utilizado. Neste caso, "None", nenhum (o primeiro registro ainda não foi utilizado).

Tabela 4.3: Comandos de controle e tela de APPEND.

Saindo da Tela e Armazenando o Arquivo

No próximo capítulo faremos algumas operações com os campos e registros que construímos. Agora, vamos sair da tela do dBASE III Plus e voltar ao *prompt* do sistema operacional para que aprendamos como desligar o computador sem perder nossos dados. (Nunca desligue o computador sem certificar-se de que todo o trabalho seja *salvo*.)

Sua tela deve estar exibindo o Registro 4 vazio. Acione o comando Ctrl-End ou Ctrl-W para sair do Registro 4 para o menu principal. (A partir do *prompt* “ . ” simplesmente digite ASSIST.)

Ctrl-End

ou

Ctrl-W

(ou ASSIST ENTER a partir do “ . ”)

A seguir selecione a subopção *Quit dBASE III* da opção *Set Up*, do menu principal ou digite

QUIT ENTER

a partir do *prompt* “ . ”.

Assim que você receber o *prompt* do DOS, faça os procedimentos já vistos para retirada do seu disquete de trabalho corretamente e em segurança. (Consulte o Capítulo 2 sobre o Sistema Operacional MS-DOS ou PC-DOS.)

No próximo capítulo utilizaremos o mesmo disquete de trabalho para executar uma série de operações importantes. Sugerimos que o usuário refaça as operações deste capítulo para maior interação com o dBASE III Plus (mas, antes de refazê-las, passe para o próximo capítulo e verifique a nova estruturação do banco de dados com seus novos nomes de campo aos quais nos referiremos por boa parte deste livro).

Capítulo

5

Manipulação e Pesquisa no Arquivo

PROCEDIMENTOS PARA USAR O MENU

Há duas maneiras de interagir com o dBASE III Plus. Pode-se acionar seus comandos através de seus Menus de Trabalho, que, quando chamados à tela, exibem suas características específicas e a partir do *prompt* (.) onde o usuário digita os comandos diretamente no teclado ao invés de selecioná-los no menu.

A partir deste capítulo abordaremos os dois modos. Entretanto, recomendamos ao usuário que, após conhecer melhor o programa, seus comandos e funções, utilize o modo a partir do *prompt*, por ser este mais prático e depender exclusivamente de seus conhecimentos de programação, cujo ensino é nosso objetivo maior.

Para sua melhor referência, as instruções equivalentes para acesso via menu estarão destacadas nos quadros sob o título “**MENU**”. As instruções voltadas para aplicação a partir do *prompt* continuarão destacadas em **negrito**.

Manipulando Nomes

Vamos começar por redefinir a estrutura de nosso banco de dados, para que você possa interagir com o livro em toda sua plenitude. Doravante usaremos um número reduzido de campos, devido ao tamanho limitado das páginas deste livro. (Você verá mais sobre modificação da estrutura do banco de dados no Capítulo 7.) Primeiro carregue seu computador com o disquete de sistema número 1 do dBASE. A seguir, carregue o disco 2 e entre no programa.

Abra seu arquivo CLIENTES.DBF com a opção *Set Up* do menu, no acionador B e, a seguir, após responder “N” à última pergunta do computador, vá para a opção *Modify*, do menu. Selecione a subopção *Database file*. Agora você já deverá estar de volta à estrutura original do seu arquivo.

Como sugestão fique somente com as seguintes definições de campo:

SNOME (para *sobrenome*)
NOME
ENDERECO
CIDADE
ESTADO
CEP

Com o cursor no primeiro campo, PROFISSÃO, acione o comando Ctrl-U para apagar os campos indesejáveis. Repare que o campo seguinte assume sua posição numérica e assim por diante. Para modificar o nome de um campo, como por exemplo de SOBRENOME para SNOME, use Ctrl-N para inserir o novo campo e continue apagando, como desejar, os campos que você não quer. Aproveite para alterar os tamanhos dos campos: use 10 para

SNOME, 10 para NOME, mantenha 25 para ENDEREÇO, use 8 para CIDADE, mantenha 2 para ESTADO e 6 para CEP. Quando terminar, saia do menu *Modify* com ESC e Y ou ENTER ENTER após o final do registro 6 e entre no modo *Append* do menu *Update*. Vamos alterar dois nomes. Use Rio ao invés de Rio de Janeiro e S. Paulo, no lugar de São Paulo, para evitar que seus nomes de campo sejam “comidos” pelo programa. Por exemplo, a palavra Rio de Janeiro num campo programado para 8 caracteres ficará assim: Rio de J; São Paulo se transformará em São Paul, e assim por diante. (Não será preciso modificar o resto, os outros campos já estão dentro dessa nova programação.)

Agora chame seu arquivo CLIENTES à tela para que nós possamos acrescentar mais alguns clientes à nossa lista. Com o dBASE no acionador A e o disquete de trabalho com seu arquivo CLIENTES no acionador B selecione.

```
MENU
Set Up
Database file
B:
CLIENTES.DBF
N
```

Após o *prompt* solicite ao dBASE:

```
USE B: CLIENTES  ENTER
```

(Lembrete: sempre pressione ENTER sobre a opção iluminada)

Analizando os Procedimentos

Vamos analisar cada procedimento, em separado, com relação aos comandos que acabamos de digitar ou selecionar. Primeiro é bom que o usuário saiba que estamos tentando *abrir* o arquivo CLIENTES.DBF que criamos no capítulo anterior. Para abrir um arquivo a partir do menu basta que destaquemos a opção *Set Up* do Menu Principal seguido da seleção da subopção *Database file*, da seleção do nome do acionador em uso (nesse caso usamos o acionador B. No caso de disco rígido selecione C), da confirmação do nome do arquivo (CLIENTES) sempre através do pressionamento da tecla ENTER. No caso de utilização do *prompt* para executarmos a mesma operação, basta que digitemos o comando USE, o nome do acionador, o nome do arquivo, e finalmente, que pressionemos a tecla ENTER. Ao final deve-se escolher se desejamos um arquivo indexado ou não, respondendo à mensagem “*is the file indexed? Y/N*” com Y (sim) ou N (não): optamos, por enquanto, pelo “N”.

Agora, utilizando os procedimentos já aprendidos com APPEND e exercitados no Capítulo 4, acrescente mais alguns clientes ao arquivo (pressione ESC para entrar no *prompt* e poder usar APPEND ou use o menu *Update*). Aceite nossas sugestões e insira

Carinca	Priscilla	R. da Cocada, 3	Miracema	RJ	04001
Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
Gavota	Isaura	R. das Aguias, 747	Rio	RJ	20000
Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100

nos novos registros 4, 5, 6, 7 e 8 respectivamente. (Caso você queira seguir os exemplos com 10 campos que sugerimos no Capítulo 4, não haverá problema algum. Simplesmente continue preenchendo os demais campos com as informações adicionais particulares que você tiver. Note, porém, que também alteramos alguns nomes de campo, como SOBRENOME, por exemplo, que, para efeito de estética, passou a ser SNOOME.) Agora vamos a uma operação de busca:

Pesquisando com LIST FOR

Vamos começar com o comando LIST FOR. Suponha que você queira uma lista das pessoas que moram na área de código CEP 04001. Você poderia selecionar:

MENU

Retrieve

List

Build a search condition

CEP

= Equal To

04001

ENTER

No more conditions

Execute the command

N

ou digitar o comando (caso queira digitá-lo você mesmo, pressione ESC agora e só depois digite)

LIST FOR CEP = '04001' ENTER

Analisando os Procedimentos

Analisando os procedimentos que acabamos de executar, separadamente, vemos que primeiro optamos por executar o comando List for a partir do menu, escolhendo a opção *Retrieve* de recuperação de dados. A seguir selecionamos a opção *List* que indica o comando que será utilizado para a operação de busca, seguido da seleção da opção *Build a search condition* que “construirá” uma determinada condição de busca a ser indicada pelo usuário na próxima seleção. A próxima seleção foi escolher o campo a ser trabalhado. Note que após a terceira seleção, *Build a search condition*, um menu com os campos que

você estruturou anteriormente apareceu na tela. Você deve então escolher o campo no qual deverá ser efetuada a operação. Nesse caso queremos trabalhar com o campo CEP. Selecione então o campo CEP. A seguir, selecione a opção = *Equal To* para indicar que você deseja igualar uma condição. Após a mensagem “*Enter a character string (without quotes):*” (insira um string de caracteres – sem aspas), digite seu *string* (nesse caso o número do CEP, 04001) e, a seguir, tecla ENTER para confirmar o comando. Antes de executar o comando é necessário selecionar a opção *No more conditions* que diz ao dBASE que você não deseja mais nenhuma outra condição. Para executar o comando que você acabou de construir, basta selecionar a opção *Execute the command* seguido da confirmação ou não de uma ordem de impressão. No nosso caso escolheremos gerar nossos dados somente na tela, portanto, selecionamos *No* para não imprimí-la. Caso você prefira selecionar as opções a partir do *prompt*, tecla ESC primeiro e então copie as instruções sugeridas.

A seguir veja os resultados de nosso comando LIST FOR:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
1	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
4	Carinca	Priscilla	R. da Cocada, 3	Miracema	RJ	04001
5	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001

Isto é, todas as pessoas com o CEP = 04001.

Podemos também exibir somente determinados campos, especificando-os com o menu (use letras maiúsculas ou minúsculas). Por exemplo:

```

MENU
Retrieve
List
Construct a field list
SNOME
NOME
ENDERECO
CEP
- >
Build a search condition
CEP
= Equal To
04001
ENTER
No more conditions
Execute the command
N

```

ou diretamente do *prompt*

```
LIST FOR CEP = '04001' SNOME, NOME, ENDERECO, CEP  ENTER
```

nos dá uma lista com somente os campos que pedimos:

Record #	SNOME	NOME	ENDERECO	CEP
1	Silva	Jose	R. da Calamidade, 13	04001
4	Carinca	Priscilla	R. da Cocada, 3	04001
5	Pinduba	Larissa	R. da Cocada, 3	04001

Analisando os Procedimentos

Esses procedimentos são parecidos com os procedimentos anteriores, exceto no que se refere à identificação dos campos que escolhemos. A seguir usamos a forma direta, isto é, a partir do *prompt*. Primeiro definimos, através do menu, as mesmas instruções com a inclusão de mais três novas seleções, ou seja, *Construct a field list* para “construirmos” uma lista dos campos que serão afetados por nossa operação, a seleção de SNOME, NOME, ENDEREÇO e CEP para indicarmos que campos são esses, e o pressionamento da tecla com a seta à direita para sairmos do menu de nomes de campo. O restante das instruções é o mesmo das instruções anteriores.

Você também pode reorganizar a ordem na qual os campos são dispostos. Se quisesse visualizar o CEP à esquerda poderia selecionar

MENU

Retrieve

List

Construct a field list

CEP

SNOME

NOME

ENDERECO

→

Build a search condition

CEP

= Equal To

04001

ENTER

No more conditions

Execute the command

N

ou digitar

LIST FOR CEP = '04001' CEP, SNOME, NOME, ENDERECO ENTER

que nos apresentaria na tela

Record #	CEP	SNOME	NOME	ENDERECO
1	04001	Silva	Jose	R. da Calamidade, 13
4	04001	Carinca	Priscilla	R. da Cocada, 3
5	04001	Pinduba	Larissa	R. da Cocada, 3

Agora o CEP aparece primeiro, de acordo com o que solicitamos. Suponha que você queira achar todas as pessoas da nossa lista que moram em Miracema. O comando selecionado do menu

MENU

Retrieve

List

Build a search condition

CIDADE

= Equal To

Miracema

No more conditions

Execute the command

N

ou o mesmo comando a partir do *prompt*

LIST FOR CIDADE = 'Miracema' ENTER

obterá o mesmo resultado, ou seja:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
4	Carinca	Priscilla	R. da Cocada, 3	Miracema	RJ	04001
5	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001

(Porém muita atenção: se você digitar miracema com “m” minúsculo, nada aparecerá nos campos pois não há nenhuma cidade chamada “miracema” e sim “Miracema”).

Analizando os Procedimentos

As instruções que acabamos de executar têm os mesmos procedimentos que as instruções LIST FOR CEP = '04001' ENTER e respectivo MENU, exceto pela escolha de CIDADE no lugar de CEP e da indicação 04001 no lugar de Miracema (e da ausência de ENTER entre Miracema e *No more conditions* no menu, porque Miracema atingiu os 8 caracteres de tamanho do campo e foi diretamente para *No more conditions*, a próxima opção).

Os números na coluna à esquerda são os números do registro, isto é, Priscilla Carinca é o quarto registro do banco de dados e Larissa Pinduba é o quinto registro. O número do registro é muito importante para a edição, como veremos posteriormente.

Se nós precisarmos de uma tela com todos os clientes cujo CEP seja “maior que” 00100, aplicaremos as seguintes instruções a partir do menu:

```

MENU
Retrieve
List
Build a search condition
CEP
> More Than
00100
ENTER
No more conditions
Execute the command
N

```

ou as instruções a partir do *prompt*

```
LIST FOR CEP > '00100' ENTER
```

e o resultado será:

Record #	SNOME	NOME	ENDereco	CIDADE	ESTADO	CEP
1	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
2	DeLanteuill	Celso	R. das Aguias, 767	Rio	RJ	20000
3	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
4	Carinca	Priscilla	R. da Cocada, 3	Miracema	RJ	04001
5	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
6	Gaivota	Isaura	R. das Aguias, 747	Rio	RJ	20000
7	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
8	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100

Analizando os procedimentos

Analizando os procedimentos dos comandos do *prompt* e do menu anterior, vemos que utilizamos a mesma sintaxe do último menu, com exceção do sinal de maior que e o número do Cep a ser comparado. Nota-se claramente que podemos utilizar quaisquer parâmetros existentes no menu (isto é, os comandos do dBASE e nossos campos) para “construir” as operações desejadas.

Especificando a Exibição dos Campos

Se desejássemos visualizar uma tela com todos os registros com CEP “menor que” 10000, solicitaríamos ao dBASE:

MENU
Retrieve
List
Construct a field list
SNAME
NAME
ENDERECO
CEP
 ->
Build a search condition
CEP
 < *Less Than*
 10000
ENTER
No more conditions
Execute the command
 N

ou os respectivos comandos do *prompt*:

LIST FOR CEP > '10000' ENTER

e obteríamos o seguinte resultado:

Record #	SNAME	NAME	ENDERECO	CEP
1	Silva	Jose	R. da Calamidade, 13	04001
3	Narcisi	Gisella	R. Borges Lagoa, 1044	04038
4	Carinca	Priscilla	R. da Cocada, 3	04001
5	Pinduba	Larissa	R. da Cocada, 3	04001
7	Pedro	Joao	R. Principal, 12	05120
8	Pedroso	Antonio	R. da Praia, 10	07100

Analisando os Procedimentos

Os procedimentos para obtenção da tela que acabamos de criar são todos nossos velhos conhecidos. Novamente usamos primeiro os comandos via menu. Depois de *Retrieve* para recuperar as informações e de *List*, o comando que lista os dados solicitados, preferimos a opção *Construct a field list* para “dizer” ao programa que iremos modificar alguma coisa na lista de campos (nesse caso pedimos ao menu para gerar somente 4 campos) seguido da seleção dos campos desejados e, conseqüentemente, da seta à direita, para abandonar os nomes de campo. A seguir, usamos os procedimentos normais para definição de nossa tarefa, ou seja, *Build a search condition* para preparar o programa para a condição de busca desejada, CEP para indicar o campo ao qual desejamos comparar uma determinada condição, o símbolo < para indicar a natureza da comparação (nesse caso, os CEP menores que 10000), seguido do rótulo pertencente ao CEP (10000) e de ENTER. Finalmente, optamos pelas condições de praxe (por enquanto), ou seja, *No more conditions* para indicar que não

desejamos mais nenhuma condição de busca, *Execute the command* para ordenar ao programa que execute o comando e *No* para dizer ao menu que não queremos imprimir nada na impressora.

Observe que, de nossos oito registros, somente seis são menores que 10000. Os registros de Celso (2) e Isaura (6) são ambos maiores, em relação ao CEP (20000).

Vejamos agora algumas técnicas adicionais com o comando LIST FOR.

Aproveite, e como exercício, acrescente mais um registro (o de número 9) ao seu banco de dados. Aceite nossa sugestão (aceite mesmo!) e insira os seguintes dados:

SNAME	NAME	ENDereco	CIDADE	ESTADO	CEP
NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000

nos campos designados. (Se errar neste ponto, é melhor voltar ao início do livro e procurar aprender tudo de novo.) Copie exatamente como propomos: “NARCISI” em maiúscula e o restante como segue).

A seguir pressione ENTER, retorne ao menu e depois acione os comandos:

```

MENU
Retrieve
List
Construct a field list
SNAME
NAME
->
Build a search condition
SNAME
= Equal To
Narcisi
ENTER
No more conditions
Execute the command
N

```

ou o equivalente a partir do *prompt*:

```
LIST FOR SNAME = 'Narcisi' SNAME, NAME ENTER
```

e você verá na tela

Record #	SNAME	NAME
3	Narcisi	Gisella

Observando nosso programa/arquivo, vemos que temos duas pessoas com o sobrenome "Narcisi" (Gisella e Anna). Então por que eles não aparecem sob o comando LIST FOR que acabamos de usar?

A resposta é que o dBASE, na realidade, não reconhece um nome, mas uma codificação, um *formato*. E o formato do sobrenome "Narcisi" de nosso registro 9 foi inserido originalmente em maiúsculas. E, como especificamos em nossa solicitação de LIST, letras minúsculas, o dBASE, acertadamente, nos mostrou o que pedimos.

Se quiséssemos exibir o sobrenome "NARCISI" também, teríamos de especificá-lo de acordo com seu formato original. Porém, se quisermos "enganar" o dBASE e listar todos os sobrenomes do arquivo de uma só vez, incluindo os em minúsculas e maiúsculas, simplesmente devemos acionar o comando

LIST UPPER(SNAME) ENTER

a partir do *prompt*, para obter o seguinte resultado:

Record #	UPPER(SNAME)
1	SILVA
2	DELANTEUIL
3	NARCISI
4	CARINCA
5	PINDUBA
6	GAIVOTA
7	PEDRO
8	PEDROSO
9	NARCISI

ou seja, todos os sobrenomes foram transformados em maiúsculas e assim nenhum sobrenome foi esquecido.

Podemos ainda, chamar todos os sobrenomes que, se transformados em maiúsculas, seriam idênticos. Por exemplo:

LIST FOR UPPER(SNAME) = 'NARCISI' ENTER

ou, alternativamente,

MENU
Retrieve
List
Build a search condition
SNAME
= Equal To
NARCISI
Combine with .OR.

SNOME
= Equal To
Narcisi
No more conditions
Execute the command
N

(Estudaremos a seguir as alternativas *.AND.* e *.OR.*)

nos mostraria todos os sobrenomes “Narcisi” de nosso arquivo, como na tela a seguir:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
3	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
9	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000

Como você deve ter notado, usamos para o exemplo anterior, a opção *Combine with .OR.* do menu, para chamarmos os dois sobrenomes idênticos – porém com formatos de letras diferentes –, que já haviam sido chamados por intermédio da instrução *LIST FOR UPPER(SNOME) = “NARCISI” ENTER.*

Usamos esses procedimentos porque o menu não contém a opção *UPPER* (por enquanto). Na realidade, o que fizemos foi pedir ao *dBASE* que nos mostrasse na tela todos os sobrenomes do campo *SNOME* que tivessem o formato “Narcisi” *ou* todos os que tivessem o formato “NARCISI”.

As opções de combinação *.AND.* e *.OR.* podem ser usadas dentro do mesmo contexto, porém por situações diferentes. Por exemplo, suponha que você queira exibir na tela todos os clientes que forem residentes no Estado do Rio *e* que tiveram o CEP 04001. Para programar essas duas condições, selecione:

MENU
Retrieve
List
Build a search condition
ESTADO
= Equal To
RJ
Combine with .AND.
CEP
= Equal To
04001
ENTER
No more conditions
Execute the command

ou seu equivalente, saindo do menu com *ESC* e ativando o *prompt* (lembre-se, para retor-

nar ao menu digite *Assist*):

LIST FOR ESTADO = 'RJ' .AND. CEP = '04001' ENTER

O resultado seria:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
4	Carinca	Priscilla	R. da Cocada, 3	Miracema	RJ	04001
5	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001

Observe que temos outros clientes em nosso banco de dados que residem no Estado do Rio, porém não têm a comparação solicitada, ou seja, o CEP igual. Note, por exemplo, que o registro de José Silva atende ao quesito CEP mas ele não reside no Estado do Rio e sim no estado hipotético IM (IMaginário), portanto só nos restou dois registros com CEP e ESTADO idênticos.

Combinando .AND., .OR. e \$

Agora queremos listar na tela, um registro que atenda a duas condições sendo que uma delas pode ser composta de um detalhe em uma outra condição própria. Complicado? Bem, vamos ao raciocínio: queremos listar na tela os clientes residentes no Rio e que tenham a palavra “Águias” ou a palavra “Canto” no campo ENDEREÇO. Para isso digite:

LIST FOR CIDADE = 'Rio' . AND. ('Águias' \$ENDERECO .OR. 'CANTO' \$ENDERECO) ENTER

e sua tela exibirá

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
3	DeLanteuil	Celso	R. das Águias, 767	Rio	RJ	20000

Compare a lógica dos dois comandos, .AND. e .OR. que acabamos de aplicar. Se for preciso, pratique um pouco com exemplos próprios até se sentir seguro suficiente para continuar. Num capítulo posterior estaremos usando essas combinações em interações mais avançadas e será interessante não termos de parar para detalhes de etapas já conquistadas.

Usando LOCATE

O comando LOCATE é útil para localizar informações no banco de dados diretamente. A diferença entre LOCATE e LIST é que LIST apresenta todo o registro na tela e LOCATE só nos dá sua posição. Como exemplo, vamos usar LIST e LOCATE num mesmo registro. Digite, a partir do *prompt*:

LIST FOR SNOME = 'DeLanteuil' ENTER

Sua tela deverá exibir:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
2	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000

Agora experimente, com a mesma sintaxe, usar LOCATE:

MENU
Position
Locate
Build a search condition
SNOME
 = *Equal To*
DeLanteuil
No more conditions
Execute the command

LOCATE FOR SNOME = 'DeLanteuil' ENTER

Sua tela deverá estar assim:

Record = 2

Para exibir o registro encontrado, digite

DISPLAY ENTER

e você verá

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
2	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000

Muita atenção. Se você digitar o nome incorretamente, como "Delanteuil" ao invés de "DeLanteuil", receberá a seguinte mensagem de erro: "*End of locate scope*" que significa que você não indicou ao dBASE a informação correta do formato do nome a ser pesquisado e/ou que a busca chegou ao fim. Por outro lado, se você quiser encontrar o registro de Pedro *ou* de Pedroso, o comando LOCATE guardará o formato dos dois nomes pois ele reconhecerá que o formato de "Pedroso" *passa* pelo formato de "Pedro" e ainda tem mais duas letras ("so"). Ou seja, "Pedroso" tem o mesmo formato de "Pedro" *comparando-se* a quantidade *total* de letras que compõem um nome inteiro e o início de outro. Caso nossa explicação esteja confusa, passe logo para a prática. Digite:

MENU
Position
Locate
Build a search condition
SNOME

= Equal To
 Pedro
 No more conditions
 Execute the command

LOCATE FOR SNAME = 'Pedro' ENTER

e o dBASE exibirá:

Record 7

que vem a ser o registro de “Pedro”. Agora acione o comando

CONTINUE ENTER

e sua tela mostrará

Record 8

Digite DISPLAY e sua tela exibirá:

Record #	SNAME	NAME	ENDereco	CIDADE	ESTADO	CEP
8	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100

ou seja o comando CONTINUE permite que possamos continuar pesquisando um nome semelhante, indefinidamente, até que todos sejam encontrados. (Poderíamos pesquisar mais de 100 nomes, por exemplo, que fossem “quase” idênticos a “Pedro” como Pedroca, Pedros, Pedrosos etc...) Mais uma coisa importante: *a ordem inversa não é verdadeira*, ou seja, o dBASE procurará pela ocorrência de “Pedroso” se você iniciar a busca com “Pedro”, mas não continuará a busca com CONTINUE caso você especifique, no início, o nome mais comprido. Experimente!

Como um suporte extra, podemos ainda usar a capacidade de *busca exata* do dBASE, ativando o comando EXACT, a partir do *prompt*.

Para ativar EXACT, acione a instrução:

SET EXACT ON ENTER

Agora você pode digitar comandos como

LOCATE FOR SNAME = 'Pedro' ENTER

e seu programa não mais procurará (mesmo com CONTINUE) pelo registro 8 de Pedroso. Neste caso sua tela exibirá:

Record 7

Em outras palavras, o comando SET EXACT ON só busca dados *exatamente* idênticos aos dados solicitados. O comando SET EXACT OFF desliga essa função.

Programando as Teclas de Função

Ainda dentro de nossa abordagem ao comando SET EXACT, vamos conhecer uma importante qualidade dessa característica, o comando SET.

O comando SET nos traz uma tela com sete opções que não serão consideradas aqui, com a exceção da opção *Options* e da opção *Keys*. A opção *Options* do menu do comando SET nos traz todos os parâmetros de SET, como por exemplo EXACT. Se você levar o cursor até *Exact* e pressionar ENTER, verá que o comando *Exact* é desativado (OFF) ou ativado (ON). No momento ficaremos por aqui quanto às explicações dos parâmetros que compõem o comando SET. O que nos importa mais agora, são as teclas de função. Pressione a tecla com a seta à direita (->) duas vezes até a opção *Keys*. Observe que sua tela exibe um quadro com 9 funções das teclas F2 a F10, inclusive. A tecla F1 não aparece porque é reservada para uma única função – F1 é a tecla Help de auxílio. As funções que aparecem no quadro são as funções *default* ou as funções que alguém programou anteriormente. Consulte o quadro de teclas de função na Figura 5.1.

F2	assist;
F3	list;
F4	dir;
F5	display structure;
F6	display status;
F7	display memory;
F8	display;
F9	append;
F10	edit;

Figura 5.1: Teclas de função do dBASE III Plus.

(A tecla F1 é reservada para Help.)

Agora leve o cursor até a tecla “F4 dir;” e pressione ENTER. Observe que uma seta aponta para o nome da função. Caso você queira mudá-la, por exemplo para “set”, digite “set;” (sem aspas) e pressione ENTER novamente. A seguir pressione ESC para retornar ao *prompt*. Agora pressione a tecla que você programou, F4, e veja o resultado. Experimente também com as outras teclas. Retorne ao *prompt* e pressione a tecla F3 (list, no programa do autor) e veja o resultado. (Lembre-se: você só pode usar as teclas de função,

com exceção de F1, a partir do *prompt*.)

É muito mais prático, por exemplo, sair do menu para o *prompt* com ESC e voltar ao menu por meio de uma tecla de função como F2 do que ter de digitar *assist* repetidamente.

No próximo capítulo trabalharemos com várias técnicas de classificação de arquivos. No Capítulo 7 abordaremos a Edição, Atualização e a Alteração de um banco de dados, e no Capítulo 8 veremos as capacidades de depuração de uma pesquisa com arquivos *Query*.

Capítulo

6

Técnicas de Classificação de Arquivos

Captar informações, armazená-las e manipulá-las não supre todas as necessidades de quem deve lidar com um banco de dados. É preciso saber organizar essas informações, classificá-las de modo a ordenar todo o trabalho já feito. E é sobre isto que falaremos neste Capítulo.

Nos capítulos anteriores, vimos como criar e acrescentar informações em um banco de dados. Vimos também como acessar e recuperar as informações desejadas através das instruções LIST e LOCATE, e dos menus específicos. Agora vamos abordar a reorganização dos registros através de sua classificação e/ou indexação.

CLASSIFICANDO COM SORT

Para classificar um banco de dados é necessário criar um novo banco de dados com os dados classificados e copiar seu conteúdo no banco de dados (ou arquivo) original. Por exemplo, com o arquivo CLIENTES.DBF em uso classifique em ordem alfabética o campo CIDADE, armazenando os dados classificados em um novo arquivo para posterior transferência para CLIENTES. Chame o novo arquivo de TRANSF (de *transferência*). Selecione:

```
MENU
Organize
Sort
CIDADE
->
B:
TRANSF
```

ou o equivalente a partir do *prompt*:

```
SORT ON CIDADE TO B:TRANSF   ENTER
```

e você verá a seguinte mensagem:

```
100% sorted      9 records sorted
(100% classificado) (9 registros classificados)
```

Não digite LIST agora. Se o fizer, sua tela não exibirá o arquivo classificado! Lembre-se, primeiro você tem de copiar seu trabalho de TRANSF para CLIENTES (ou então listar TRANSF). Para fazer isto, abra o novo arquivo TRANSF.DBF com *Set Up, Database file, B: TRANSF.DBF, N* para copiá-lo em CLIENTES. A seguir acione as instruções

```
MENU
Organize
Copy
B:
CLIENTES
Execute the command
```

ou, através do *prompt*:

COPY TO B:CLIENTES ENTER

A mensagem

CLIENTES.DBF already exists, overwrite it? (Y/N)

(CLIENTES.DBF já existe, gravar sobre ele?)

aparece na tela solicitando autorização do usuário para sobregravar (Y) ou não (N) o arquivo já existente. Tecla Y (sim) pois seu antigo arquivo CLIENTES não contém os dados alfabetizados. Após Y, o arquivo CLIENTES assumirá a classificação do arquivo TRANSF e exibirá a mensagem *9 records copied*. Pressione qualquer tecla, chame CLIENTES à tela (LIST ou, através do menu, *Retrieve, List, Execute the command, N*) e você verá:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
1	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
2	Carinca	Priscilla	R. da Cocada, 3	Miracema	RJ	04001
3	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
4	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
5	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
6	Gaivota	Isaura	R. das Aguias, 747	Rio	RJ	20000
7	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
8	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
9	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100

ou seja, todos os campos reclassificados de acordo com a ordem alfabética do campo CIDADE.

Agora vamos classificar os clientes de nosso conhecido arquivo CLIENTES.DBF, pelo campo NOME. Tudo o que temos a fazer é repetir a operação anterior, com exceção do campo indicado e, no caso de ainda estarmos com o arquivo (TRANSF) aberto, mudarmos o nome do arquivo de apoio (ou *fecharmos*¹ TRANSF – neste exemplo, preferimos mudar seu nome).

Selecione os comandos

```

MENU
Organize
Sort
NOME
->
B:
TRANSF2

```

ou o equivalente no *prompt*

¹ Para fechar um arquivo use o comando USE sem parâmetros ou CLOSE DATABASE.

SORT ON NOME TO B:TRANSF2 ENTER

A seguir pressione qualquer tecla e copie o conteúdo de TRANSF2 no arquivo CLIENTES. Abra TRANSF (você pode usar as seleções do menu já aprendidas, ou usar USE a partir do *prompt*).

Quando você se certificar de que TRANSF2.DBF está aberto, acione a instrução

MENU*Organize**Copy**B:**CLIENTES**Execute the command*

ou

COPY TO B:CLIENTES

para copiar o novo conteúdo de TRANSF2 em CLIENTES. Não se esqueça de teclar Y em resposta à solicitação de permissão para sobregravar CLIENTES.

Para visualizar novamente seu arquivo CLIENTES atualizado, você terá de abrí-lo e executá-lo através do comando USE ou das instruções de menu sobre as quais já mencionamos, ou seja (para lembrar...)

MENU*Set Up**Database file**B:**CLIENTES.DBF**ENTER**N*

ou

USE B:CLIENTES ENTER

Agora CLIENTES.DBF está pronto para uso com sua nova classificação dos nomes de todos os clientes.

Para ver o arquivo na tela ainda temos de terminar as instruções. Acione o comando

MENU*Retrieve*

List
Execute the command
N

ou, utilizando o *prompt*

LIST ENTER

e todo o arquivo, reclassificado, aparecerá na tela:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
3	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Gaivota	Isaura	R. das Aguias, 747	Rio	RJ	20000
6	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
7	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
8	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
9	Carinca	Priscilla	R. da Cocada, 3	Miracema	RJ	04001

Agora vamos supor que queiramos fazer uma classificação pelo campo “CEP”, em ordem decrescente (/D)². Isto é fácil para o dBASE. Simplesmente digite o comando, a partir do *prompt* (se você estiver no menu, tecle ESC):

SORT ON CEP/D TO B:TEMP ENTER

e o dBASE emitirá a mensagem

100% sorted 9 records sorted
(100% classificados) (9 registros classificados)

Para visualizar os dados de sua nova tela, acione o comando

USE B:TEMP ENTER

para abrir o arquivo. Após o *prompt*,

COPY TO B:CLIENTES ENTER

para que nosso arquivo CLIENTES.DBF fique atualizado com essa nova organização. A seguir, após teclar Y à solicitação de regravação, digite:

USE B:CLIENTES ENTER

para colocar CLIENTES.DBF, no acionador B, em uso e

² /D não está disponível para operações com o menu.

LIST **ENTER** (ou F3, se preferir)

para, finalmente, visualizá-lo:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
2	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
3	Gaivota	Isaura	R. das Aguias, 747	Rio	RJ	20000
4	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
6	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
7	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
8	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
9	Carinca	Priscilla	R. da Cocada, 3	Miracema	RJ	04001

Se preferir usar o menu para visualizar sua tela:

MENU
Set Up
Database file
B:
TEMP.DBF
ENTER
N

para abrir seu arquivo. A seguir, para copiá-lo em CLIENTES.DBF, siga o seguinte procedimento:

MENU
Organize
Copy
B:
CLIENTES.DBF
Execute the command
Y

Abra CLIENTES.DBF:

MENU
Set Up
Database file
B:
CLIENTES.DBF
N

Agora estamos prontos para listá-lo:

MENU*Retrieve**List**Execute the command**N*

E teremos a tela com a classificação por ordem decrescente de Cep da página 60.

Pratique mais um pouco com o comando SORT. Use os procedimentos que você já aprendeu e, diretamente do *prompt* (lembre-se, para sair do menu tecle Esc), ou através do menu Assistente, classifique os dados novamente pela ordem alfabética do campo NOME. Deixe o arquivo CLIENTES.DBF *exatamente* como estava *antes* da classificação decrescente pelo campo CEP. Nas próximas seções iremos trabalhar com CLIENTES no formato da página 59. Boa sorte!

APAGANDO OS DADOS (A PARTIR DO PROMPT (.))

Para apagar ou *deletar* uma informação, temporária ou permanentemente, no dBASE III Plus, o usuário precisa de executar duas operações. Isto torna-se positivo, apesar de um pouco mais demorado do que se houvesse uma só operação, por razões de segurança.

O dBASE primeiro nos faz ter a certeza de que queremos apagar a informação selecionada obrigando-nos a visualizá-la com as *marcas de deleção* antes de confirmar o comando e apagá-las, irreversivelmente, do arquivo. Por exemplo, para apagar o último registro de nosso arquivo CLIENTES, o Registro 9, primeiro devemos marcá-lo com DELETE. Saia do menu assistente com Esc e, a partir do *prompt* (.), acione o comando

DELETE RECORD 9 ENTER

que nos trará a mensagem

1 record deleted

(1 registro apagado)

(Caso você não saiba qual o número do registro, chame a condição necessária, através de LIST FOR ou do menu. No próximo capítulo veremos como deletar registros através do menu.)

Bem agora chame o arquivo novamente à tela para visualizá-lo com a marca (*) de deleção. Use os procedimentos já conhecidos, ou seja:

LIST ENTER (ou simplesmente, F3)

ou, caso você esteja no modo Assist,

MENU*Retrieve**List**Execute the command**N*

Agora veja a tela com a marca de deleção (*):

Record #	SNOME	NOME	ENDEREÇO	CIDADE	ESTADO	CEP
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
3	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Gaivota	Isaura	R. das Aguias, 747	Rio	RJ	20000
6	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
7	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
8	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
9	*Carinca	Priscilla	R. da Cocada, 3	Miracema	RJ	04001

Observe que nossa cliente Priscilla Carinca está pronta para ser eliminada de nosso arquivo.

Para prosseguir com nossa operação de deleção, acione o comando

PACK ENTER

e o dBASE responderá com a mensagem

8 records copied
(8 registros copiados)

avisando que somente 8 registros, dos nove originais, foram mantidos no arquivo.

Se listarmos o arquivo agora, não veremos Priscilla Carinca.

Record #	SNOME	NOME	ENDEREÇO	CIDADE	ESTADO	CEP
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
3	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Gaivota	Isaura	R. das Aguias, 747	Rio	RJ	20000
6	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
7	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
8	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001

Vamos ver o que acontece se deletarmos um nome no meio do arquivo. Por exemplo, o registro 5.

Acione o comando

DELETE RECORD 5 ENTER

1 record deleted
(1 registro apagado)

PACK ENTER

Chame o arquivo à tela e veja o que aconteceu. Use o comando

LIST ENTER

ou o menu

MENU
Retrieve
List
Execute the command
N

(ou simplesmente, F3)

Agora veja a tela:

Record #	SNOME	NOME	ENDereco	CIDADE	ESTADO	CEP
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
3	DeLanteuil	Celso	R. das Aguas, 767	Rio	RJ	20000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
6	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
7	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001

Observe que todos os registros abaixo de Isaura Gaivota reorganizaram sua ordem numérica assumindo sua sequência lógica. O dBASE está sempre preocupado com a organização de seus arquivos.

Mas o que acontece se desistirmos de apagar permanentemente um registro? Poderíamos recuperá-lo? Bem, recuperar um registro marcado para deleção só é possível *antes* de executarmos o comando (*fatal*) PACK. Por exemplo, marque para deleção o registro 3.

DELETE RECORD 3 ENTER

Agora liste o arquivo na tela.

LIST ENTER

(ou, através do menu)

MENU*Retrieve**List**Execute the command**N*

Veja a tela com a marca de deleção.

Record #	SNAME	NOME	ENDERECO	CIDADE	ESTADO	CEP
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
3	*DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
6	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
7	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001

A seguir, vamos anular o processo de deleção com o comando

RECALL RECORD 3 ENTER

Com outro LIST, vemos

Record #	SNAME	NOME	ENDERECO	CIDADE	ESTADO	CEP
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
3	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
6	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
7	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001

ou seja, Celso DeLanteuil não está mais marcado para ser apagado do arquivo CLIENTES.DBF. Nesse caso, o comando PACK não teria mais efeito.

Podemos também executar deleções globais, usando o comando DELETE ALL FOR *condição*. Se quiséssemos deletar, por exemplo, todos os clientes do Estado do Rio de Janeiro, poderíamos simplesmente digitar o comando:

DELETE ALL FOR ESTADO = 'RJ' ENTER

e o dBASE responderá com

3 records deleted
(3 registros apagados)

Listando o arquivo veríamos

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
1	*NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
3	*DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
6	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
7	*Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001

ou seja, todos os indivíduos que residem no Estado do Rio de Janeiro estão marcados para deleção e posterior eliminação do arquivo.

Como não queremos ficar com somente quatro clientes em nosso banco de dados (os que moram em outros estados que não RJ), vamos cancelar a ordem de apagamento. Digite

RECALL ALL ENTER

e todos os sete registros anteriores (menos os 2 que já apagamos antes) aparecerão. Liste o arquivo novamente e veja o resultado:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
3	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
6	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
7	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001

Todos os clientes estão de volta normalmente.

Evitando a Perda Acidental

Um bom dispositivo de segurança para evitar o apagamento acidental de dados é listar somente os registros que foram marcados para deleção. Por exemplo, marque para deleção todos os clientes residentes no Estado do Rio. Use o comando:

DELETE ALL FOR ESTADO = 'RJ' ENTER

e o dBASE exibirá a resposta

3 records deleted
(3 registros apagados)

Dessa forma anulamos três clientes. Agora, liste somente o que foi marcado para anulação.
_ Acione o comando

LIST FOR DELETED () ENTER

e você verá

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
1	*NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
3	*DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
7	*Pinduba	Larissa	R. da Cocada, 5	Miracema	RJ	04001

Para não perder seus registros permanentemente, digite o comando

RECALL ALL ENTER

(Caso você queira apagá-los, já sabe: é só digitar **PACK ENTER** e seus registros terão ido “para o espaço”).

Apagando Todo o Banco de Dados

Para terminar, vamos aprender a apagar todo um arquivo indesejável. Como exemplo, vamos apagar nosso arquivo **TRANSF.DBF** que criamos anteriormente para ajudar a classificar o banco de dados **CLIENTES.DBF**. Cuidado, ao apagar o arquivo **TRANSF**, nós o perderemos permanentemente. Use o comando

ERASE B:TRANSF.DBF ENTER

e você receberá a mensagem

File has been deleted
(Arquivo foi apagado)

confirmando que **TRANSF.DBF** foi permanentemente apagado de nossa memória. (Aproveite, faça o mesmo com **TEMP.DBF**.)

No próximo capítulo introduziremos as técnicas de apagamento de registros via menu.

CLASSIFICANDO COM INDEX

Podemos executar classificações mais eficiente e rapidamente com o comando **INDEX** do que com **SORT**. Com **INDEX** não precisamos criar outro arquivo para ordenar os dados e depois passá-los para o arquivo original como fizemos com **TRANSF** e **CLIENTES**; em outras palavras, não precisamos usar o comando **COPY**. Por exemplo,

INDEX ON SNOME TO B:NOVOS ENTER

ou seu equivalente no menu

MENU*Organize**Index**SNOME**B:**NOVOS**ENTER*

(Durante sua interação com o menu, o dBASE exibirá duas mensagens. A primeira explicando que você pode usar qualquer caractere ou expressão numérica ou de datas, envolvendo um ou mais campos no arquivo de banco de dados. Você deve obedecer à ordem final *“Enter an index key expression.”* com o campo no qual deseja trabalhar – SNOME, nesse caso. A segunda mensagem pede que o usuário insira um nome de arquivo com até 8 letras ou dígitos, seguido de um ponto e uma extensão com até três letras ou dígitos. Digite o nome do novo arquivo à mensagem final *“Enter the name of the file.”*. Como você está trabalhando com o comando INDEX, não será preciso usar a extensão do nome de arquivo. O dBASE se encarrega de acrescentar a extensão .NDX ao arquivo recém-criado. Você não precisa ler estas mensagens. Simplesmente selecione as opções do menu na ordem vertical apresentada terminando com ENTER.)

Agora, liste o arquivo e veja os resultados.

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
3	DeLanteuil	Celso	R. das Agulas, 767	Rio	RJ	20000
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
7	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
6	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001

Os registros estão na ordem alfabética por sobrenomes (do campo SNOME) mas seus números de identificação permanecem os mesmos. E mais, não foi preciso usar o comando COPY para transferência de arquivo como acontece com SORT.

NOVOS.NDX, o novo arquivo recém-criado, será um subarquivo do arquivo CLIENTES, contendo os sobrenomes classificados de CLIENTES.DBF. Sempre que você desejar consultar alguma informação sobre determinado cliente, basta consultar seu “índice” e ver o número de seu registro:

DeLanteuil	3
NARCISI	1
Narcisi	4
Pedro	5
Pedroso	2
Pinduba	7
Silva	6

Chame seu arquivo CLIENTES.DBF agora com a característica de indexação (o subarquivo NOVOS.NDX). Use o comando

USE B:CLIENTES INDEX ON B:NOVOS ENTER

ou o equivalente do menu

```

MENU
Set Up
Database file
B:
CLIENTES.DBF
Y
NOVOS.NDX
->

```

Analizando os Procedimentos

Para abrir o arquivo CLIENTES.DBF, primeiro selecionamos a opção *Set Up* do menu principal (o *Assistant*); a seguir selecionamos a opção *Database file* que especifica que queremos trabalhar com um arquivo de banco de dados; em seguida selecionamos o acionador onde se encontra nosso arquivo para depois identificá-lo. No final respondemos *sim* (Y) à pergunta sobre se o arquivo é indexado e selecionamos o nome do arquivo indexado (com as setas verticais e a tecla ENTER sobre as opções selecionadas, no caso de mais de uma opção. Para completar, tecle “->” (a seta à direita).

Para visualizar seu arquivo indexado, liste CLIENTES.DBF:

LIST ENTER

ou

```

MENU
Retrieve
List
Execute the command
N

```

e sua tela deverá estar assim:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
3	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120

2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
7	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
6	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001

Mudando a Ordem de Apresentação dos Campos

Suponha que você queira classificar seu arquivo pela ordem do campo ESTADO. Vamos supor também que você queira, para melhor visualização, exibir o campo ESTADO na primeira coluna, à esquerda no lugar original da coluna SNAME. Vejamos os procedimentos.

Primeiro acione o comando

INDEX ON ESTADO TO B:CODIGO ENTER

ou

MENU
Organize
Index
ESTADO
B:
CODIGO
ENTER

A seguir liste o arquivo na ordem desejada.

LIST ESTADO,SNAME,NOME,ENDereco,CIDADE,CEP ENTER

ou o equivalente do menu

MENU
Retrieve
List
Construct a field list
ESTADO
SNAME
NOME
ENDereco
CIDADE
CEP
 ->
ENTER
Execute the command
N

e você verá o seguinte resultado:

Record #	ESTADO	SNOME	NOME	ENDERECO	CIDADE	CEP
2	BA	Pedroso	Antonio	R. da Praia, 10	Salvador	07100
6	IM	Silva	Jose	R. da Calamidade, 13	Ficticia	04001
1	RJ	NARCISI	Anna	R. do Canto, 7	Rio	21000
3	RJ	DeLanteuil	Celso	R. das Aguias, 767	Rio	20000
7	RJ	Pinduba	Larissa	R. da Cocada, 3	Miracema	04001
5	RS	Pedro	Joao	R. Principal, 12	P. Alegre	05120
4	SP	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	04038

Os registros agora aparecem com sua ordem original alterada, com ESTADO em primeiro plano e em ordem alfabética. (Caso sua tela não se pareça com a do exemplo, analise seus procedimentos anteriores. Você deve ter se esquecido de algum detalhe importante, ainda que pequeno.)

Vantagens de INDEX Sobre SORT

A principal vantagem do comando INDEX sobre o comando SORT é sua velocidade. Um arquivo de índice (.NDX) ocupa menos espaço na memória. Outra vantagem é que, com INDEX, pode-se acrescentar dados ao programa, ou alterá-lo sem que seja necessário repetir todo o procedimento de inicialização do comando novamente. Entretanto, um cuidado especial deve ser tomado se quisermos manter *todos* os nossos arquivos .NDX atualizados: mantê-los *ativos*. Isto significa na prática que, se quiséssemos chamar (abrir) nosso arquivo CLIENTES.DBF para qualquer alteração (com o comando APPEND) ou apenas para visualização, teríamos de seguir os seguintes procedimentos:

```
USE B:CLIENTES INDEX B:NOVOS,B:CODIGO    ENTER
```

ou

```
MENU
Set Up
Database file
B:
CLIENTES.DBF
Y
NOVOS.NDX
CODIGO.NDX
->
```

que, em outras palavras, quer dizer: “*abra* o arquivo CLIENTES.DBF e *ative* os arquivos indexados NOVOS.NDX *primeiro*, mas não se esqueça de ativar *também* o arquivo CODIGO.NDX”.

Agora, experimente o comando LIST:

LIST ENTER

ou

MENU

Retrieve

List

Execute the command

N

A seguir veja o resultado:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
3	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
7	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
6	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001

Isto significa dizer que abrimos o arquivo CLIENTES e ativamos *os dois* arquivos indexados que construímos anteriormente, mas a ordem de apresentação é liderada pelo primeiro arquivo comandado (NOVOS.NDX). Também significa que, se quiséssemos fazer qualquer alteração em CLIENTES.DBF, os arquivos NOVOS.NDX e CODIGO.NDX, por estarem ambos ativos, também seriam automaticamente atualizados. (Caso esquecêssemos de incluir qualquer um dos arquivos indexados como, por exemplo, em

USE B:CLIENTES INDEX B:NOVOS ENTER

onde não incluímos CODIGO.NDX, NOVOS.NDX seria automaticamente atualizado, mas CODIGO.NDX teria de ser criado novamente, desde o início.)

Podemos também ativar os dois arquivos de índice, NOVOS e CODIGO, com o comando SET INDEX, *a partir do prompt*. Por exemplo:

SET INDEX TO B:NOVOS, B:CODIGO ENTER

Este comando nos permitiria exibir os dados na ordem contida no arquivo de índice NOVOS.NDX (ou seja, o campo SNOME classificado) mas também ativaria o arquivo de índice CODIGO.NDX. Ambos seriam automaticamente atualizados caso o usuário desejasse alterar o arquivo principal CLIENTES.DBF.

Vamos exibir agora CLIENTES.DBF pela ordem do campo ESTADO, ou seja, com CODIGO.NDX em primeiro plano:

USE B:CLIENTES INDEX B:CODIGO,B:NOVOS ENTER

ou, através do menu

```

MENU
Set Up
Database file
B:
CLIENTES.DBF
Y
CODIGO.NDX
NOVOS.NDX
->

```

para chamar o arquivo. Agora, para exibí-lo

```
LIST  ENTER
```

ou

```

MENU
Retrieve
List
Execute the command
N

```

e o resultado seria o seguinte:

Record #	SNOME	NOME	ENDEREÇO	CIDADE	ESTADO	CEP
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
6	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
3	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
7	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038

mas o arquivo NOVOS.NDX (que contém os sobrenomes em ordem alfabética) ainda permanece ativo.

Note também que o arquivo .NDX principal (o primeiro a ser selecionado), por indicar *prioridade* na ordem da exibição, é chamado de *Master*. (Neste livro nos referiremos a ele como *principal* mesmo.)

Reindexando Arquivos

Para recriar arquivos de índice que foram desativados pode-se usar o procedimento inicial, por exemplo:

```
USE B:CLIENTES
INDEX ON SNAME TO B:NOVOS    ENTER
INDEX ON ESTADO TO B:CODIGO  ENTER
```

ou seu equivalente no menu

```
MENU
Set Up
Database file
B:
CLIENTES.DBF
ENTER
N
```

} para abrir CLIENTES.DBF

```
MENU
Organize
Index
SNAME
B:
NOVOS
ENTER
Y
```

} para indicar, pelo campo SNAME, o arquivo NOVOS.NDX

```
MENU
Organize
Index
ESTADO
B:
CODIGO
ENTER
Y
```

} para indexar, pelo campo ESTADO, o arquivo CODIGO.NDX

ou *reindexar* os arquivos de índice desativados (porém existentes):

```
USE B:CLIENTES INDEX B:NOVOS, B:CODIGO    ENTER
REINDEX    ENTER
```

ou

```
MENU
Set Up
Database file
B:
CLIENTES.DBF
```

} para “abrir” CLIENTES.DBF

Y	}	para ativar os arquivos indexados
NOVOS.NDX		
CODIGO.NDX		
->	}	para sair do submenu

e, a seguir, digitar, a partir do *prompt*:

REINDEX ENTER } para reindexar NOVOS e CODIGO.

Limitações

Pode-se trabalhar com até sete arquivos indexados por vez. Porém não se esqueça de que as atualizações que eventualmente forem feitas no programa somente surtirão efeito se os arquivos indexados estiverem ativos.

CLASSIFICAÇÕES MÚLTIPLAS

As classificações múltiplas, ou seja, as classificações dentro de classificações, são muito necessárias quando se deseja “amarrar” mais de um campo ao mesmo tempo, como nome e sobrenome de determinados clientes. Por exemplo, suponha que você precise dos nomes e sobrenomes de seu arquivo CLIENTES classificados por ordem alfabética. Para obter o resultado desejado, você terá de *concatenar* os campos a serem indexados com um sinal de mais (+). Como prática, acione o comando, a partir do *prompt*,

INDEX ON SNAME + NOME TO B:COMPLETO ENTER

ou, a partir do *Assistant*,

```

MENU
Organize
Index
SNAME + NOME
ENTER
B:
COMPLETO

```

Analisando os Procedimentos

Primeiro chamamos (com a seta ou teclando a primeira letra) a opção principal **Organize**. A seguir selecionamos o comando *Index* (subopção de **Organize**) para mostrar ao programa que queremos trabalhar com a característica de indexação. Continuando, digitamos (após a mensagem) os nomes dos dois campos que serão utilizados como referência de nossa

indexação, separados pelo sinal de mais (+). Pressionamos ENTER a seguir, o nome do acionador que será utilizado na operação e, finalmente, o nome do arquivo que armazenará os dois campos indexados – COMPLETO.NDX (isto é, o arquivo COMPLETO conterá todo o nosso arquivo CLIENTES sendo que os campos SNAME *mais* NOME estarão re-classificados).

Liste o arquivo e veja o resultado: Digite

LIST ENTER (ou F3)

MENU

Retrieve

List

Execute the command

N

e sua tela deverá estar assim:

Record #	SNAME	NOME	ENDERECO	CIDADE	ESTADO	CEP
3	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
7	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
6	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001

Agora, usando os procedimentos já vistos anteriormente, acrescente mais três registros ao seu arquivo *original* CLIENTES.DBF. Caso você prefira usar o menu, selecione a opção *Update* seguida da subopção *Append*. Ao receber sua “tela-formulário em branco”, preencha-a com as seguintes informações cadastrais:

SNAME	NOME	ENDERECO	CIDADE	ESTADO	CEP
Narcisi	Helena C.	R. do Canto, 7	Rio	RJ	21000
Pinduba	Lilica	R. da Cocada, 3	Miracema	RJ	04001
Pedro	Joca	R. Reinprecht, 100	S. Paulo	SP	04038

Ao terminar o último registro, tecle ENTER em vez de preencher mais outro sobrenome.

Agora CLIENTES.DBF deve estar pronto para uso, com 10 registros. Vamos recuperá-lo para confirmar. (Não se esqueça de abrir – USE – CLIENTES.DBF, caso você ainda não o tenha feito!). Digite:

LIST ENTER

ou

MENU

Retrieve

*List**Execute the command**N*

e você verá a seguinte tela:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
3	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
6	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
7	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
8	Narcisi	Helena C.	R. do Canto, 7	Rio	RJ	21000
9	Pinduba	Lilica	R. da Cocada, 3	Miracema	RJ	04001
10	Pedro	Joca	R. Reinprecht, 100	S. Paulo	SP	04038

Agora vamos alterar nosso arquivo novamente. Vamos classificá-lo pela ordem de ESTADO, SNOME e NOME:

INDEX ON ESTADO + SNOME + NOME TO B:COMPLETO ENTER Y

ou

MENU*Organize**Index**ESTADO+SNOME+NOME**ENTER**B:**COMPLETO**ENTER**Y*

Liste seu arquivo e você verá a seguinte tela:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
2	Pedroso	Antonio	R. da Praia, 10	Salvador	BA	07100
6	Silva	Jose	R. da Calamidade, 13	Ficticia	IM	04001
3	DeLanteuil	Celso	R. das Aguias, 767	Rio	RJ	20000
1	NARCISI	Anna	R. do Canto, 7	Rio	RJ	21000
8	Narcisi	Helena C.	R. do Canto, 7	Rio	RJ	21000
7	Pinduba	Larissa	R. da Cocada, 3	Miracema	RJ	04001
9	Pinduba	Lilica	R. da Cocada, 3	Miracema	RJ	04001
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120
4	Narcisi	Gisella	R. Borges Lagoa, 1044	S. Paulo	SP	04038
10	Pedro	Joca	R. Reinprecht, 100	S. Paulo	SP	04038

Analisando a tela você pode perceber que o dBASE concatenou todos os registros colo-

cando-os em ordem alfabética *no conjunto*, ou seja, respeitando primeiro a ordem do campo ESTADO, depois a ordem do campo SNAME e, a seguir, a ordem do campo NOME.

Lembre-se, você pode listar somente os campos que desejar ver, evitando assim um consumo desnecessário de tempo. Para isso você pode digitar o comando a partir do *prompt* ou selecionar os procedimentos a partir do menu.

Como exemplo (mais uma vez) vamos usar os campos que acabamos de indexar, ESTADO, SNAME e NOME. Digite:

LIST ESTADO, SNAME, NOME ENTER

ou, a partir do menu:

MENU

Retrieve

List

Construct a field list

ESTADO

SNAME

NOME

->

Execute the command

N

Nesse caso teríamos a seguinte tela (com o campo ESTADO na primeira coluna, como pedimos):

Record #	ESTADO	SNAME	NOME
2	BA	Pedroso	Antonio
6	IM	Silva	Jose
3	RJ	DeLanteuil	Celso
1	RJ	NARCISI	Anna
8	RJ	Narcisi	Helena C.
7	RJ	Pinduba	Larissa
9	RJ	Pinduba	Lilica
5	RS	Pedro	Joao
4	SP	Narcisi	Gisella
10	SP	Pedro	Joca

Busca Rápida de Registros com FIND e Display

Para acelerar a busca de um registro em um banco de dados extenso, onde LIST e LOCATE levariam alguns segundos ou minutos para processar, podemos usar o comando FIND com um arquivo indexado sempre com base no arquivo indexado principal.

Digite, a partir do *prompt*:

USE B:CLIENTES INDEX B:NOVOS, B:CODIGO ENTER³

ou, como sempre

```

MENU
Set Up
Database file
B:
CLIENTES.DBF
Y
NOVOS.NDX
CODIGO.NDX
->

```

Agora, somente a partir do *prompt*, digite⁴

FIND 'Pedro' ENTER

ou, se preferir o menu:

```

MENU
Position
Seek
'Pedro'
ENTER
X

```

Analizando os Procedimentos

A partir da primeira opção, Position, chegamos à seleção do comando SEEK, o equivalente de FIND do menu. A seguir definimos o que queremos encontrar e o digitamos entre aspas caso a informação seja do tipo caractere (se for do tipo numérico, por exemplo, não coloque aspas). Para confirmar o comando, digitamos ENTER e, finalmente, pressionamos qualquer tecla (aqui usamos “X” para representar uma tecla qualquer) para retornarmos ao menu para a próxima introdução (DISPLAY).

Pronto, seu registro com os dados de João Pedro já foi encontrado. Para exibi-lo, simplesmente digite

DISPLAY ENTER

³ Você não precisa identificar o acionador (B:) se no início definir que seus arquivos já estarão no acionador. Use o comando SET DEFAULT TO B: ENTER antes do comando USE.

⁴ Lembre-se, para usar comandos a partir do *prompt*, você tem de sair do modo Assist, através da tecla Esc.

ou

MENU

Retrieve

Display

Execute the command

ENTER

e você verá o registro:

Record #	SNOME	NOME	ENDERECO	CIDADE	ESTADO	CEP
5	Pedro	Joao	R. Principal, 12	P. Alegre	RS	05120

Mais Velocidade de Busca

Neste Capítulo aprendemos a usar os comandos SEEK (do menu) e Find (do *prompt*). Vale aproveitar este resto de capítulo para enfatizar o importante detalhe da velocidade de operação desses comandos – muito superior à de seus equivalentes, principalmente em se tratando de arquivos muito longos. A construção **LIST WHILE SNOME = “Narcisi”**, caso seu arquivo tivesse mais de 1000 registros com sobrenomes idênticos e seus arquivos indexados estivessem ativos), seria muito mais rápida do que a construção **LIST FOR SNOME = “Narcisi”**. Lembre-se também de que essas instruções podem ser executadas também a partir dos mesmos *Retrieve, List, Build a scope condition, SNOME, = Equal To, Narcisi, No more conditions, Execute the command* e *Retrieve, List, Build a search condition, SNOME, = Equal To, Narcisi, No more conditions, Execute the command*, respectivamente. Experimente usar SEEK ou FIND com seu arquivo ou, preferivelmente, com um arquivo extenso.

No próximo capítulo trataremos da edição de um banco de dados, sua atualização e alterações; no Capítulo 8 trataremos dos arquivos do tipo *Query* e da depuração dessa pesquisa; no Capítulo 9 abordaremos, em tempo, os números, as datas e as variáveis de memória.

Capítulo

7

**Edição do Banco de Dados
Atualização e Alteração**

Suponha que você queira alterar seu banco de informações, por exemplo, trocar alguns campos ou “enxugar” o arquivo. Neste capítulo vamos *editar* nosso programa, ou seja, vamos precisar somente de algumas informações úteis para trabalhar, como nome, sobrenome e telefone de nossos clientes. Para isso temos de chamar CLIENTES.DBF e alterar sua estrutura.

Primeiro vamos chamar CLIENTES.DBF e apagar, *permanentemente*, os campos ENDE-REÇO, CIDADE, ESTADO e CEP, para depois incluir o novo campo TEL.

A partir do *prompt* (com CLIENTES aberto), digite

MODIFY STRUCTURE ENTER

ou, a partir do menu

MENU

Modify

Database file

e a estrutura de seu banco de dados aparecerá assim:

Bytes remaining: 3939

CURSOR < - - - > Char: ↔ Word: Home End Pan: ^ ← ^ →	INSERT Char: Ins Field: ^ N Help: F1	DELETE Char: Del Word: ^ Y Field: ^ U	Up a field: ↑ Down a field: ↓ Exit/Save: ^ End Abort: Esc
---	--	--	--

	Field Name	Type	Width	Dec
1	<u>SNOME</u>	Character	10	
2	NOME	Character	10	
3	ENDERECO	Character	25	
4	CIDADE	Character	8	
5	ESTADO	Character	2	
6	CEP	Character	6	

MODIFY STRUCTURE: < B: > CLIENTES :Field: 1/6 :Ins :
Enter the field name.

Field names begin with a letter and may contain letters, digits and underscores

Repare que o cursor apresenta-se na mesma posição, ou seja, sob a letra “S” do campo SNAME. Caso você queira modificar o nome, tipo ou tamanho de um campo, simplesmente escreva por sobre o campo já existente, mas muita atenção: se você deseja mudar, por exemplo, o nome do campo e o seu tamanho, simultaneamente, isto não será possível numa mesma operação. Você terá de acionar MODIFY STRUCTURE (ou *Modify Database*

se file) uma vez para os nomes de campo e *outra*, complementar, para seus tamanhos. Além disso, para confirmar as alterações, você terá de pressionar Ctrl-W ou Ctrl-End e sair do modo MODIFY STRUCTURE deixando o programa com suas novas definições. (Para cancelar a operação e permanecer com as definições anteriores, pressione Esc ou Ctrl-Q – *antes de confirmá-las!*) E o mais importante: só é possível alterar os nomes de campo de uma estrutura *vazia*, isto é, que não contiver nomes, sobrenomes e endereços *preenchidos* no arquivo, sob o risco de se perder tudo!

Inserindo Novos Campos

Temos duas formas principais de modificar, com segurança, a estrutura de nosso banco de dados: uma é acrescentar mais um campo ao final dos campos já existentes, isto é, após o campo CEP, acrescentaríamos o campo TEL; a outra maneira seria inserirmos entre campos já existentes, o campo que gostaríamos de ter. Por exemplo, para acrescentar agora o campo TEL após o campo CEP, simplesmente pressione a tecla com a seta para baixo (↓) seis vezes (ou tecle Ctrl-X) até o novo registro, o registro 7. É só escrever o novo campo, TEL e preencher suas definições: **Type** – *Character* (não confunda número *telefônico* com números *aritméticos*. O número telefônico não será usado para cálculos de nenhum tipo – multiplicar o número telefônico de um cliente pelo de outro cliente não faz nenhum sentido). A seguir, após definir o campo TEL como “*Character*”, defina o tamanho do campo como: **Width** – 14.

Para salvar a nova estrutura com seu arquivo agora constando de 7 registros, digite ^W ou ^End (Ctrl-W ou Ctrl-End). A mensagem

Press ENTER to confirm – any other key to resume

Database records will be APPENDED from backup fields of the same name only!!

(Pressione ENTER para confirmar – qualquer outra tecla para interromper. Os registros somente serão atualizados com nomes de campos originais!!)

aparecerá em seguida. Pressione ENTER.

Inserção no Meio do Arquivo

Para acrescentar o novo campo TEL entre outros campos já existentes, primeiro apague o novo campo 7 com Ctrl-U, leve, com a seta para baixo, o cursor para o local onde deseja inserir o novo campo. Aqui vamos inserir e preencher TEL logo após o campo NOME, ou seja, TEL terá de ser o campo número 3, portanto, com o cursor em ENDEREÇO, digite Ctrl-N e você terá o espaço necessário para inserir seu novo campo, TEL como no exemplo a seguir:

	Field Name	Type	Width	Dec
1	SNAME	Character	10	
2	NOME	Character	10	
3	TEL	Character	14	
4	ENDERECO	Character	25	
5	CIDADE	Character	8	
6	ESTADO	Character	2	
7	CEP	Character	6	

Eliminando Campos

Agora, após confirmar seu trabalho com Ctrl-End e ENTER, usando o último exemplo, vamos executar a operação definitiva, ou seja, apagar os campos 7, 5 e 4. Para executar esta operação, leve o cursor até o campo que será apagado, como no exemplo a seguir (você terá de acionar MODIFY STRUCTURE novamente):

	Field Name	Type	Width	Dec
1	SNAME	Character	10	
2	NOME	Character	10	
3	TEL	Character	14	
4	ENDERECO	Character	25	
5	CIDADE	Character	8	
6	ESTADO	Character	2	
7	CEP	Character	6	

Com o comando Ctrl-U (^U na tela de informações acima), apague todo o campo CEP e os campos superiores ESTADO, CIDADE e ENDEREÇO. Com Ctrl-End e ENTER salve todo o trabalho. Sua tela de campos deverá estar assim antes de retornar para o menu ou para o *prompt*:

	Field Name	Type	Width	Dec
1	SNAME	Character	10	
2	NOME	Character	10	
3	TEL	Character	14	

Lembre-se, nunca mude o nome de um campo *preenchido* pois se o fizer, você perderá todas as informações contidas nele. Use os procedimentos comentados na pág. 83 para alterar campos vazios. A razão dessa limitação está no fato de o dBASE criar, automaticamente, uma nova área de trabalho, cada vez que tentamos modificar sua estrutura. Esta área de trabalho é, na realidade, um *banco de dados paralelo* usado para *compor* a nova estrutura, enquanto o banco de dados atual é apagado. Uma vez que o banco de dados paralelo esteja preenchido, ele é gravado sobre o banco de dados atual com as novas informações.

Finalmente, antes de entrarmos no mérito dos comandos EDIT e BROWSE, vamos analisar

os comandos de MODIFY STRUCTURE (ou *Modify Database file*).

Teclas com as setas	Movem o cursor para a direção indicada
BACKSPACE	Deleta o caractere à esquerda do cursor
Ctrl-D ou →	Cursor à direita
Ctrl-E ou ↑	Cursor para cima
Ctrl-X ou ↓	Cursor para baixo
Ctrl-S ou ←	Cursor à esquerda
Del	Deleta o caractere sobre o cursor
Ctrl-M ou ENTER	Move o cursor para o próximo item
Ctrl-N	Insere um novo campo na posição do cursor
Ctrl-Q ou Esc	Sai do programa sem salvar os dados
Ctrl-U	Deleta o campo na posição do cursor
Ctrl-V ou Ins	Ativa (e desativa) o modo de inserção
Ctrl-W ou Ctrl-End	Sai do programa e salva as alterações
Ctrl-Y	Deleta a palavra na posição do cursor

Tabela 7.1: Comandos de Controle do Cursor de MODIFY STRUCTURE

Usando EDIT

A edição ou *editoração*, é uma característica muito importante em um banco de dados. Um arquivo – ou cadastro – de clientes de uma determinada empresa ou atividade, necessita de ser atualizado com certa frequência. Endereços e telefones, por exemplo, mudam com mais frequência do que os nomes dos clientes em determinadas atividades, ao passo que para outras, muda-se mais de cliente (quando então tem de se alterar todo um registro).

Para trabalhar com EDIT você terá de ter seu arquivo aberto, inclusive os arquivos de índice, para que as atualizações aconteçam automaticamente.

Podemos também editar via menu ou via comando direto. Neste próximo exemplo, iremos editar nossos registros 1 e 4. Acione o comando

EDIT 1 ENTER

ou, via menu,

MENU

Position

Locate

Build a search condition

SNOME

= Equal To

NARCISI
ENTER
Combine with AND.
NOME
= Equal To
Anna
No more conditions
Execute the command

para definir, previamente, o registro que será editado. Ou, usando o número do registro,

MENU
Position
GoTo Record
RECORD
1
ENTER

e a seguir

MENU
Update
Edit

e depois visualize o resultado:

SNOME	:Narcisi	:
NOME	:Anna	:
TEL	:	:

Pronto! Agora já podemos editar nosso cadastro do cliente Anna Narcisi. Mova o cursor para o campo TEL e digite o número telefônico da Sra. Narcisi. Por exemplo,

TEL: (021)645-2525_:

Caso você queira alterar alguma informação do cadastro, como, por exemplo, colocar outro nome, pressione Ctrl-Y que apagará todo o conteúdo *à direita* do campo indicado. (Veja adiante o conjunto de teclas de controle do cursor na Tabela de Comandos de EDIT.) Ctrl-Y no modo EDIT é uma boa opção para alterar endereços de seu banco de dados, caso você tenha de atualizar suas fichas com certa frequência.

Uma outra opção interessante é o comando SET MENU. Com SET MENU ON, por exemplo, ativa-se um menu de funções das teclas de controle do cursor para os modos EDIT, BROWSE, APPEND e MODIFY STRUCTURE. Com SET MENU OFF (experimente) desativa-se o menu. Porém, vale lembrar que isto só é necessário com comandos diretos do *prompt*. Para comandos de menu já estão incluídos um resumo das relações de teclas de

controle sobre as quais acabamos de mencionar.

Para salvar a(s) alteração(ões) feita(s), tecle Ctrl-W ou Ctrl-End e o programa armazenará os novos dados do registro 1 de Anna Narcisi e retornará ao *prompt*.

BACKSPACE	Deleta o caractere à esquerda do cursor
Ctrl-D ou →	Cursor à direita
Ctrl-E ou ↑	Cursor para cima
Ctrl-X ou ↓	Cursor para baixo
Ctrl-S ou ←	Cursor à esquerda
Ctrl-G ou Del	Deleta o caractere sobre o cursor
Ctrl-T	Deleta uma palavra à direita
Ctrl-Y	Deleta todo o conteúdo do campo na posição do cursor
Ctrl-U	Deleta todo o registro
Ctrl-V ou Ins	Ativa (e desativa) o modo de inserção
Ctrl-R ou PgUp	Retrocede um registro
Ctrl-C ou PgDn	Avança um registro
Ctrl-End ou Ctrl-W	Salva as alterações e retorna ao menu ou ao <i>prompt</i>
Ctrl-Q ou Esc	Abandona as alterações e retorna ao menu ou ao <i>prompt</i>

Tabela 7.2: Comandos de Controle do Cursor de EDIT

Agora, vamos acrescentar o número telefônico de todos os clientes com sobrenome DeLanteuil. Primeiro chame seu arquivo (caso ainda não o tenha chamado). A seguir digite

LIST FOR SNAME = 'DeLanteuil' ENTER

ou, a partir do menu

```

MENU
Retrieve
List
Build a search condition
SNAME
= Equal To
DeLanteuil
No more conditions
Execute the command
N

```

o que nos dará a seguinte tela:

Record #	SNAME	NOME	TEL
3	DeLanteuil	Celso	

ou seja, só temos um cliente com esse sobrenome em nosso banco de dados. Mas já sabemos em que registro ele se encontra e agora podemos editá-lo. Acione o comando

EDIT 3 ENTER

junto ao *prompt* do dBASE III Plus ou use os procedimentos do menu

MENU

Update

Edit

e você terá a tela de campos do registro 3:

Record No.	3
SNAME	: DeLanteuil :
NOME	: Celso :
TEL	: :

Preencha o número telefônico (por exemplo, (021)067-7074) e salve as alterações com Ctrl-W ou Ctrl-End.

Ainda há uma outra maneira de acessar um determinado registro (conhecido), via menu ou a partir do *prompt*, o comando GOTO. Para acessar diretamente um determinado registro, por exemplo, o registro 4, acione o comando

GOTO 4 ENTER

a partir do *prompt*, ou

MENU

Position

GoTo Record

RECORD

4

ENTER

a partir do menu. Para editar o registro de Gisella Narcisi basta selecionar Edit. Pode-se também visualizar o registro na tela através do comando *Browse* do menu *Update* ou ainda *Display* do menu *Retrieve*. Pode-se ainda “saltar” vários registros via comando SKIP. A opção *Skip* do menu *Position* permite que saltemos, por exemplo, 6 registros: acione

SKIP 6 ENTER

ou

MENU
Position
SKIP
6
ENTER

Experimente agora voltar um registro. Use o comando SKIP-1:

SKIP-1 ENTER

ou

MENU
Position
SKIP
-1
ENTER

e você estará no registro 9.

Usando BROWSE

A principal diferença entre EDIT e BROWSE é a capacidade “panorâmica” de BROWSE. Com este comando, pode-se movimentar a tela do dBASE horizontal e verticalmente, possibilitando uma visão global do arquivo para edição ou inserção de dados.

Pode-se acessar o modo BROWSE – que contém um submenu próprio –, via menu e via comando direto. Por exemplo, digite, diretamente do *prompt*,

GO TOP ENTER

para começar do início (topo) do banco de dados. A seguir digite o comando

BROWSE ENTER

ou, via menu, os mesmos procedimentos:

MENU
Position
GoTo Record
TOP

e, a seguir

MENU*Update**Browse*

e você terá sua tela com todas as informações *que couberem* nela:

SNOME ---	NOME ----	TEL -----
NARCISI	Anna	(021)645-2525
Pedroso	Antonio	
DeLanteuil	Celso	(021)067-7074
Narcisi	Gisella	
Pedro	Joao	
Silva	Jose	
Pinduba	Larissa	
Narcisi	Helena C.	
Pinduba	Lilica	
Pedro	Joca	

Até aqui tudo bem, todas as informações couberam na tela. Agora vamos supor que queiramos exibir com BROWSE o arquivo anterior, aquele que tinha os campos SNOME, NOME, ENDEREÇO, CIDADE, ESTADO e CEP, que apagamos há pouco. Bem, aqui aprendemos outra lição: a de nunca apagar dados que não estejam em outro arquivo – de nome diferente –, ou de nunca trabalhar com um só disquete (este sim, muito importante), precavendo-se sempre com um disquete de reserva ou *backup*, se preferir.

Se você estiver usando este livro para realmente aprender o dBASE, propomos que, num outro disquete ou no mesmo – se tiver espaço – (não faça nada ainda. Veja em “Fechando o Banco de Dados”), você crie um arquivo nos moldes do arquivo CLIENTES.DBF, com as seguintes características: *nome do arquivo* – MEFERREI.DBF (por ter apagado quase tudo do outro arquivo); *estrutura* – idêntica à de CLIENTES.DBF mas com o campo TEL logo após NOME; *classificação* – use os mesmos clientes da tela que mostramos na página 77 e dê um SORT por SNOME para que sua tela final de MEFERREI.DBF fique assim (Não se esqueça de copiar, durante a classificação com SORT, o campo SNOME em outro arquivo de nome diferente, para, só então, copiá-lo via COPY no novo arquivo MEFERREI.DBF. Mais adiante trataremos do menu *Tools* que traz opções de renomeação e cópia de arquivos.); após o comando BROWSE:

Record #	SNOME	NOME	TEL	ENDERECO	CIDADE
ESTADO	CEP				
	1	DeLanteuil Celso	(021)067-7074	R. das Aguias, 767	Rio
RJ	20000				
	2	NARCISI Anna	(021)645-2525	R. do Canto, 7	Rio
RJ	21000				
	3	Narcisi Helena C.		R. do Canto, 7	Rio
RJ	21000				
	4	Narcisi Gisella		R. Borges Lagoa, 1044	S. Paulo
SP	04038				
	5	Pedro Joao		R. Principal, 12	P. Alegre
RS	05120				
	6	Pedro Joca		R. Reinprecht, 100	S. Paulo
SP	04038				

	7	Pedroso	Antonio	R. da Praia, 10	Salvador
BA	07100				
	8	Pinduba	Larissa	R. da Cocada, 3	Miracema
RJ	04001				
	9	Pinduba	Lilica	R. da Cocada, 3	Miracema
RJ	04001				
	10	Silva	Jose	R. da Calamidade, 13	Ficticia
IM	04001				

FECHANDO O BANCO DE DADOS

Não se esqueça de alguns detalhes importantes. Você *pode* criar esse arquivo no mesmo disquete se quiser e depois fazer um *backup*. Para isso, não se esqueça de *fechar* o arquivo da mesma forma que você o abriu com o comando USE. Este procedimento é muito importante pois ele lhe garante que os dados de seu programa não se perderão facilmente, caso haja alguma perda de energia no computador, ou caso alguém retire subitamente o disquete do acionador sem avisar. Portanto, se você tenciona usar o mesmo disquete para criar seu novo arquivo MEFERREI.DBF, que será idêntico a CLIENTES.DBF, terá de fechar este último sob ameaça de perdê-lo totalmente.

Capacidade de Arquivos Ativos – As Áreas de Trabalho

O dBASE III Plus pode trabalhar com até 15 arquivos abertos de cada vez. Isto significa que há *áreas de trabalho* específicas designadas para cada arquivo que abrimos. Por exemplo, para abrirmos um arquivo na primeira área de trabalho lógica, usamos o comando SELECT, diretamente do *prompt*:

```
SELECT A      ENTER
USE CLIENTES  ENTER
```

e teremos aberto CLIENTES.DBF. Se quisermos abrir nosso novo arquivo na segunda área de trabalho, digitamos

```
SELECT B      ENTER
USE MEFERREI  ENTER
```

Caso não definamos nenhuma área específica, o programa usará a área A primeiro, por *default*.

Para fechar todos os bancos de dados podemos usar o comando CLEAR ALL (que libera arquivos, variáveis, etc.) diretamente do *prompt*. Também podemos abrir um outro arquivo em uma mesma área e, automaticamente, fechar qualquer arquivo aberto nesta mesma área. Podemos ainda fechar um arquivo em uma área de trabalho específica *sem* abrir nenhum outro arquivo, com o comando USE sem nome de arquivo.

Para fechar todos os bancos de dados sem liberar os arquivos ou variáveis, use o comando

CLOSE DATABASES ENTER

Para fechar todos os arquivos de um determinado tipo, use o comando mais a definição do tipo de arquivo. Por exemplo,

CLOSE INDEX ENTER

fecharia todos os arquivos de índice. Mas, **ATENÇÃO**: para fechar um *determinado* arquivo de banco de dados, não aplique o comando **CLOSE nome do arquivo** (como **CLOSE CLIENTES**, por exemplo). Acione o comando **USE** sem especificação do arquivo.

Agora, vamos trabalhar com nosso arquivo **MEFERREI.DBF**. Vamos chamá-lo com **BROWSE** para podermos operar com sua tela:

GO TOP ENTER
BROWSE ENTER

ou

MENU
Position
GoTo Record
TOP
<-
MENU
Update
Browse

e agora temos nossa tela **BROWSE** limitada pelo espaço físico do vídeo:

SNOME ---	NOME ----	TEL -----	ENDERECO-----	CIDADE ----	ESTADO
DeLanteuil	Celso	(021)067-7074	R. das Aguias, 767	Rio	RJ
NARCISI	Anna	(021)645-2525	R. do Canto, 7	Rio	RJ
Narcisi	Helena C.		R. do Canto, 7	Rio	RJ
Narcisi	Gisella		R. Borges Lagoa, 1044	S. Paulo	SP
Pedro	Joao		R. Principal, 12	P. Alegre	RS
Pedro	Joca		R. Reinprecht, 100	S. Paulo	SP
Pedroso	Antonio		R. da Prais, 10	Salvador	BA
Pinduba	Larissa		R. da Cocada, 3	Miracema	RJ
Pinduba	Lilica		R. da Cocada, 3	Miracema	RJ
Silva	Jose		R. da Calamidade, 13	Ficticia	IM

Isto significa que agora você pode fazer as edições que quiser. Caso você tenha usado o menu para chamar **BROWSE**, poderá usar o pequeno menu superior para referência das teclas de controle. Caso contrário acione o comando **SET MENU ON** e, no modo do

prompt, você também terá sua tabela de controle.

Movimento Horizontal da Tela

Note porém que com BROWSE podemos movimentar a tela vertical e horizontalmente via Ctrl – → ou Ctrl-B, para ver o campo CEP (e *esconder* o campo SNOME) e via Ctrl – <– ou Ctrl-Z para ver de volta o campo SNOME. Para movimentar-se por entre os campos *visíveis*, use as teclas HOME e END (ou Ctrl-A e Ctrl-F) para movimentos à esquerda e à direita, respectivamente e, no início de cada registro, para cima e para baixo.

Movimento Vertical da Tela

Para movimentar-se exclusivamente por entre os registros, utilize as setas verticais (↑ e ↓) ou Ctrl-E e Ctrl-X, respectivamente. Para movimentar-se de tela-em-tela, use as teclas PgUp e PgDn.

Para sua melhor referência, visualize as funções de controle do cursor na tabela 7.3.

Ctrl-D ou →	Cursor à direita
Ctrl-E ou ↑	Cursor para cima
Ctrl-X ou ↓	Cursor para baixo
Ctrl-S ou ←	Cursor à esquerda
Ctrl-A ou Home	Movimenta o cursor um campo à esquerda
Ctrl-F ou End	Movimenta o cursor um campo à direita
Ctrl-B ou Ctrl- →	Avança um campo à direita
Ctrl-Z ou Ctrl- ←	Avança um campo à esquerda
Ctrl-G ou Del	Deleta o caractere sobre o cursor
Ctrl-Y	Deleta dados à direita do cursor no campo
Ctrl-V ou Ins	Entra no modo INSERT para que novos dados possam ser inseridos no campo sem apagar os dados existentes
Ctrl-Home	Exibe/Apaga a tela Help
Ctrl-W ou Ctrl-End	Salva os dados editados e retorna ao prompt
Ctrl-Q ou Esc	Retorna ao prompt sem efetuar as alterações/edições

Tabela 7.3: Comandos de Controle do Cursor de BROWSE

A tela Help de BROWSE pode ser ativada com Ctrl-Home:

Bottom Top Lock Record No. Freeze Find

Bottom	Destaca o último registro do banco de dados
Top	Destaca o primeiro registro do banco de dados
Lock	“Trava” o conteúdo exibido de um ou mais campos à esquerda da tela
Record No.	Solicita o número de um registro a ser destacado (iluminado pelo cursor)
Freeze	“Trava” um campo destacado para facilitar a edição do mesmo campo em vários registros

Tabela 7.4: Opções da Tela Help de BROWSE

Usando CHANGE e REPLACE: Edição Global

Edição global significa alterar mais de um registro no banco de dados. É uma ótima opção para a execução rápida de operações extensas. Aqui usaremos um exemplo simples, considerando que temos um banco de dados muito simples – MEFERREI.DBF. Vamos alterar nossos registros da cidade do Rio de Janeiro, de Rio para Rioclaro, um nome hipotético. Façamos essa alteração como sempre: a partir do *prompt* e via menu.

Simplesmente digite, se a partir do *prompt*

REPLACE CIDADE WITH 'Rioclaro' FOR CIDADE = 'Rio' ENTER

ou, se via menu

```

MENU
Update
Replace
CIDADE
Rioclaro
->
Build a search condition
CIDADE
= Equal To
Rio
ENTER
No more conditions
Execute the command

```

e você receberá a seguinte mensagem na tela:

3 records replace
(3 registros substituídos)

Para visualizar o novo arquivo, use o comando

LIST SNAME, NOME, CIDADE ENTER

diretamente do *prompt* ou; via menu

MENU
Retrieve
List
Construct a field list
SNAME
NOME
CIDADE
Execute the command
N

Agora é só visualizar sua nova tela de MEFERREI que foi resumida, pois não precisamos de todos os campos:

Record #	SNAME	NOME	CIDADE
1	DeLanteuil	Celso	Rioclaro
2	NARCISI	Anna	Rioclaro
3	Narcisi	Helena C.	Rioclaro
4	Narcisi	Gisella	S. Paulo
5	Pedro	Joao	P. Alegre
6	Pedro	Joca	S. Paulo
7	Pedroso	Antonio	Salvador
8	Pinduba	Larissa	Miracema
9	Pinduba	Lilica	Miracema
10	Silva	Jose	Ficticia

Como você pode ver (espero que possa mesmo), todos os nomes do campo CIDADE que se chamavam Rio agora se chamam Rioclaro. Faça este exercício novamente, e transforme tudo de novo em Rio.

Outra coisa que já deve estar lhe incomodando é o sobrenome de nossa cliente Anna NARCISI que, desde o início, vem aparecendo em letras maiúsculas. Com REPLACE podemos resolver isto de uma vez por todas:

REPLACE SNAME WITH 'Narcisi' FOR SNAME = 'NARCISI' ENTER

ou, via menu

MENU
Update
Replace
SNAME

Narcisi
ENTER
 ->
Build a search condition
SNAME
 = *Equal To*
NARCISI
ENTER
No more conditions
Execute the command

Listando a tela, podemos ver o resultado. Comande uma exibição parcial, somente dos campos SNAME e NOME. A partir do *prompt*, digite

LIST SNAME, NOME ENTER

ou, a partir do menu, selecione

MENU
Retrieve
List
Construct a field list
SNAME
NOME
 ->
Execute the command
N

e sua tela deverá estar assim:

Record #	SNAME	NOME
1	DeLanteuil	Celso
2	Narcisi	Anna
3	Narcisi	Helena C.
4	Narcisi	Gisella
5	Pedro	Joao
6	Pedro	Joca
7	Pedroso	Antonio
8	Pinduba	Larissa
9	Pinduba	Lilica
10	Silva	Jose

Agora todos os sobrenomes estão escritos corretamente.

Podemos também, a partir do *prompt*, executar mudanças globais mais lentas, uma de cada vez, com o comando **CHANGE FIELD**. A diferença entre **CHANGE** e **REPLACE** é que

CHANGE não está disponível no menu, e o resultado de sua aplicação não é automático, isto é, é preciso *preencher* os dados que devemos alterar, diretamente em cada registro, até que todos os itens solicitados estejam reeditados ou corrigidos. Com REPLACE, além de podermos selecioná-lo via menu e diretamente do *prompt*, podemos executar operações automáticas sem a necessidade de confirmações individuais. Podemos também utilizar a característica abrangente de REPLACE ALL, a partir do *prompt*, para operarmos em todos os *campos* de um determinado registro ou REPLACE NEXT 3, por exemplo, que alteraria somente os próximos 3 registros do arquivo; a partir da posição atual do cursor. Vamos experimentar o comando CHANGE:

CHANGE FIELD CEP FOR CIDADE = 'Rio' ENTER

O comando acima nos apresentaria todos os registros, *um de cada vez*, cuja cidade relativa ao CEP fosse o Rio. O primeiro registro apareceria primeiro, para alteração, e assim sucessivamente. Experimente. Mude os CEPs do Rio e depois mude-os novamente para os números originais.

APAGANDO ARQUIVOS VIA MENU

No capítulo anterior vimos como apagar registros e arquivos com comandos a partir do *prompt*. Agora vamos terminar nosso treinamento de deleção de registros com seleções via menu. Como exemplo vamos chamar nosso arquivo MEFERREI.DBF. Melhor ainda, vamos aproveitar e renomear este arquivo cujo nome, tenho certeza, não é lá muito sugestivo. Para renomear seu arquivo, primeiro feche MEFERREI.DBF com USE sem argumentos. A seguir, digite a partir do *prompt*.

RENAME B:MEFERREI.DBF TO B:MEDEIBEM.DBF

(não é necessário digitar B: caso você já esteja operando no acionador B)

Caso lhe seja necessário, existem duas formas de definir o acionador *default*; uma delas é usar o comando direto do *prompt*.

SET DEFAULT TO B

para definir o acionador B: como o acionador *default*. Outra seria usando o próprio menu. Como vamos usar nosso próximo exemplo de RENAME via menu, sugiro, como prática, que usemos também a opção de atribuição de acionador (ou *drive*, se preferir) do próprio menu. Primeiro chame a opção *Tools* (e siga as instruções a seguir):

MENU

Tools

Set drive

B:

Agora siga essas instruções para finalmente renomear seu arquivo (caso você ainda não o tenha feito):

```
MENU
Tools
Rename
B:
MEFERREI.DBF
MEDEIBEM.DBF
ENTER
```

Seu arquivo MEFERREI.DBF agora chama-se MEDEIBEM.DBF, um nome bem mais sugestivo para alguém que deseja aprender através da aquisição de um livro!

Bem, como dissemos anteriormente, vamos apagar parcialmente nosso arquivo MEDEIBEM.DBF via menu. Podemos apagar registros de nosso banco de dados – marcando-os para deleção com o asterisco – usando a combinação Ctrl-U via opções *Edit* ou *Browse*.

Vamos apagar os registros referentes a Celso DeLanteuil e Antonio Pedroso. Primeiro abra seu novo arquivo (ao renomeá-lo, seu antigo arquivo foi fechado). A seguir use uma dessas duas opções:

```
MENU
Update
Edit
```

ou

```
MENU
Update
Browse
```

Mas, muita atenção: para usar o modo *Edit* via menu você deve seguir os procedimentos de definição de registro antes de selecionar *Update*, como vimos nas páginas 86. No caso de *Browse*, vá primeiro ao topo do arquivo com *Position*, *GoTo*, *TOP*. Outra diferença básica é que o comando *Edit* exhibe todo o registro na tela e o comando *Browse* só exhibe a quantidade de campos que cabem na tela.

A seguir marque o registro número 1 de seu arquivo MEDEIBEM.DBF com Ctrl-U. Marque também o registro 7. Salve sua operação com Ctrl-W ou Ctrl-End. Chame sua tela com *Retrieve*, *List*, *Execute the command*, *N* para ver os registros marcados com os asteriscos. Para apagar os registros 1 e 7 *permanentemente*, utilize a opção *Pack* do menu *Update*. O dBASE exibirá a mensagem

```
8 records copied
(8 registros copiados)
```

mostrando que agora somente 8 registros foram gravados dos dez anteriores. Liste seu arquivo, parcialmente, digamos, pelos campos SNAME e NAME, com as opções:

```
MENU
Retrieve
List
Construct a field list
SNAME
NAME
->
Execute the command
N
```

Agora você deverá ter a seguinte tela:

Record #	SNAME	NAME
1	Narcisi	Anna
2	Narcisi	Helena C.
3	Narcisi	Gisella
4	Pedro	Joao
5	Pedro	Joca
6	Pinduba	Larissa
7	Pindubã	Lilica
8	Silva	Jose

Para apagar registros a partir do *prompt*, refira-se ao Capítulo 6. Para apagar todo o arquivo use o comando `ERASE nome do arquivo.extensão` ENTER a partir do *prompt*.

No próximo capítulo discutiremos o fascinante comando *Query* que permite que selecionemos somente arquivos com determinada condição especificada previamente pelo usuário, ao invés de ter de chamá-los para então transformá-los. No Capítulo 9 trabalharemos com cálculos aritméticos, com datas e variáveis de memória. No Capítulo 10, criaremos e manipularemos bancos de dados múltiplos com o dBASE III Relacional.

Capítulo

8

**Usando Arquivos Query
Depurando Sua Pesquisa**

Neste capítulo criaremos condições especiais para recuperação e visualização de um arquivo. O comando *Query*, do menu *Create*, será utilizado para facilitar a manutenção de nosso novo arquivo. Como você pode notar, criaremos um novo arquivo, o arquivo VENDAS.DBF que será composto de uma lista de produtos vendidos por nossa empresa. A seguir começaremos a operar com uma nova forma de classificação – muito mais rápida e eficiente do que as técnicas que conhecemos até agora (mas que não dispensa as interações simples até agora praticadas com o dBASE III Plus).

Primeiro, vamos criar o arquivo VENDAS com os conhecimentos já adquiridos.

Não se esqueça de que alguns caracteres não podem ser usados como nome de campo. Por exemplo, o ponto (.) é um deles, portanto não tente abreviar “Quantidade” como QTDE., pois tudo o que acontecerá será um sinal sonoro de aviso de que seu nome não foi aceito. Tente QTDE sem o ponto e tudo ficará correto.

Para estruturar seu novo arquivo, use 4 caracteres para o tamanho (*width*) do campo CLIENTE (*Character*); 10 caracteres para o tamanho do campo PRODUTO (*Character*); 3 caracteres para o tamanho do campo QTDE (*Numeric*); 6 caracteres para o tamanho do campo OTN (*Numeric*) – incluindo 2 para a coluna *Dec* (*Decimal*); 8 caracteres para o campo VENC (*Date*) que, como é previamente definido como um campo de data, já recebe automaticamente o *default* de 8, e 1 caractere para o campo STATUS (*Logical*). Um pequeno detalhe: a coluna *Dec*, de decimal, só aceita um número no mínimo com a metade do valor do número inteiro. Isto quer dizer que se colocássemos 3 no campo OTN, a coluna *Dec* equivalente só aceitaria 1 e assim por diante.

Vamos supor que nosso novo arquivo fique assim depois de pronto:

Record	#	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
	1	001	LOTUS T.A.	40	4.00	07/20/88	.T.
	2	002	WordLegal	60	15.00	07/20/88	.T.
	3	003	WordLegal	20	15.00	07/20/88	.T.
	4	003	dBASE III	15	6.00	07/20/88	.T.
	5	001	APPLE	10	4.00	07/18/88	.T.
	6	010	ASMIBM PC	80	6.00	07/18/88	.T.
	7	042	ASMIBM PC	25	6.00	07/17/88	.T.
	8	009	LOTUS T.A.	8	4.00	07/12/88	.T.
	9	033	WordStar	80	4.00	07/08/88	.T.
	10	012	MKT	80	4.00	06/30/88	.F.
	11	005	MKT	10	4.00	06/30/88	.F.
	12	008	MATFIN	80	4.00	06/22/88	.F.
	13	014	WordLegal	40	15.00	06/22/88	.F.
	14	021	dBASE III	20	6.00	06/18/88	.F.
	15	022	dBASE III	10	6.00	06/18/88	.F.
	16	038	WordLegal	30	6.00	06/12/88	.F.
	17	037	ASMIBM PC	5	6.00	06/08/88	.F.
	18	001	dBASE III	80	6.00	06/08/88	.F.
	19	007	WordLegal	20	4.00	06/06/88	.F.
	20	019	WordLegal	60	6.00	06/04/88	.F.

Analizando o Novo Arquivo – VENDAS.DBF

O campo **CLIENTE** do nosso novo arquivo, contém uma relação de 20 clientes hipotéticos que contribuiram (será?) com o faturamento de nossa empresa imaginária, até a data de hoje; o campo **PRODUTO** relaciona o tipo de produto que vendemos (ou *pensamos* que vendemos); o campo **QTDE** mostra a quantidade de produtos vendidos para a empresa codificada no campo **CLIENTE**; o campo **OTN** exibe o preço *unitário* do produto em Obrigações do Tesouro Nacional; o campo **VENC** mostra os respectivos vencimentos de cada compra e o campo **STATUS** exibe a situação de cada um: .T. para contas *pagas* e .F. para saldo *devedor*.

Agora, com o banco de dados em uso (aberto), entre no modo **QUERY** com o comando

CREATE QUERY DEVEDOR ENTER

a partir do *prompt*, ou

```
MENU
Create
Query
B:
DEVEDOR
ENTER
```

a partir do menu.

O menu *Query* oferece quatro operações para você trabalhar: **Set Filter**, **Nest**, **Display** e **Exit**. Primeiro trabalharemos com o submenu **Set Filter**, que será usado para *filtrar* as condições especificadas em seu banco de dados **VENDAS.DBF**, criando para isso, um subprograma que aqui chamaremos de **DEVEDOR**.

Uma característica importante desse menu são suas “janelas” de dados. À medida em que vamos trabalhando com **Set Filter** e suas opções, somos verdadeiramente “presenteados” com todas as informações referentes ao arquivo principal, tais como nomes de campo, operadores matemáticos etc., que facilitam a evolução de nossas interações com a tela e, conseqüentemente, o teclado. Vejamos como tudo funciona. Selecione as opções

```
MENU
Set Filter
Field Name
STATUS
Operator
Is False
Constant/Expression
->
```

->

->

Exit**Save**

Analizando os Procedimentos

No primeiro menu, selecionamos a opção **Create** para criar uma condição de pesquisa no modo **Query**. Em seguida confirmamos nossa solicitação indicando **Query** como a operação desejada. Chamamos nosso novo arquivo – o arquivo especial que conterá somente determinadas condições do arquivo principal VENDAS.DBF –, de DEVEDOR, e confirmamos tudo com ENTER.

No menu seguinte, optamos pela opção **Set Filter**, para “filtrar” as condições boas e considerar somente as condições a serem pré-determinadas. A seguir selecionamos a opção **Field Name** para designarmos o nome do campo que contribuirá com a condição seguido da seleção do campo STATUS, o campo escolhido. Em seguida selecionamos a opção **Operator** que nos apresenta seu menu de operadores e escolhemos o operador “**Is False**”. Continuando no menu **Set Filter**, a subopção **Constant/Expression**, que já representa nossa condição e, pressionando “->” três vezes, vamos direto para o Menu **Exit** onde então selecionamos **Save** e salvamos o programa (falaremos sobre a subopção **Connect** mais adiante).

Agora temos a seguinte condição literal em DEVEDOR

STATUS is false

ou seja, todos os registros de nosso banco de dados atual que estiverem com esse “status”, serão exibidos toda vez que chamarmos o subarquivo DEVEDOR. Sua tela deverá estar como na Figura 8.1.

Analizando a tela vemos que já preenchemos a linha 1 com a condição especificada (todos os registros que apresentarem um saldo *devedor* aparecerão quando chamarmos DEVEDOR. Vamos tentar?). Chame DEVEDOR à tela agora. Acione o comando

MENU**Set Up****Query****B:****DEVEDOR**

e depois

MENU**Retrieve**

Set Filter Nest Display Exit 02:15:10 am

Field Name	STATUS
Operator	is
Constant/Expression	False
Connect	
Line Number	1

Line	Field	Operator	Constant/Expression	Connect
1	STATUS	Is	False	
2				
3				
4				
5				
6				
7				

Figura 8.1: Tela *Query* com os primeiros comandos.

List

Execute the command

N

Sua tela deverá estar assim.

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
10	012	MKT	80	4.00	06/30/88	.F.
11	005	MKT	10	4.00	06/30/88	.F.
12	008	MATFIN	80	4.00	06/22/88	.F.
13	014	WordLegal	40	15.00	06/22/88	.F.
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
17	037	ASMIBM PC	5	6.00	06/08/88	.F.
18	001	dBASE III	80	6.00	06/08/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.

Cada arquivo *Query* pode armazenar até 7 linhas de condições. Isto significa que além da condição especificada de *Devedor*, podemos acrescentar mais condições como por exemplo listar todos os registros que atendam às condições de *Devedor* e de determinado produto. Por exemplo, vamos colocar na linha 2 de nosso arquivo *Query* a segunda condição, a de que os produtos especificados sejam o *WordLegal*.

Para dar continuidade ao nosso trabalho, uma vez que já salvamos toda a operação da linha 1 e retornamos ao menu (ou ao *prompt*) com *Exit* e *Save*, digite

MODIFY QUERY DEVEDOR

ou selecione as opções no menu

MENU
Modify
Query
B:
DEVEDOR

Destaque (iluminando com a seta) a opção *Connect* do menu *Set Filter*, para que possamos “conectar” essa condição (constante) com a segunda condição. Selecione a opção *Combine with AND.*, ou seja, desejamos as seguintes condições: “que o arquivo DEVEDOR, subarquivo do arquivo principal VENDAS, contenha todos os Devedores e aqueles registros que contêm o produto WordLegal em suas transações”. A seguir siga as seguintes instruções ainda no menu *Set Filter*:

MENU
Set Filter
Field Name
PRODUTO
Operator
= Matches
Constant/Expression
“WordLegal”

No momento já completamos nossa segunda linha. Caso você não tenha mais nenhuma condição a especificar, finalize a sessão com

MENU
Exit
Save

Você já sabe como chamar, com *Modify Query*, seu arquivo *Query DEVEDOR* para futuras edições. Pode-se também utilizar *Browse* e *Edit*, do menu *Update*, para edições via chamada do arquivo. Por exemplo, os comandos

MENU
Set Up
Query
B:
DEVEDOR

MENU
Update
Browse

ativam o arquivo DEVEDOR e entram no modo BROWSE.

Cancelando Um Arquivo Query

Para cancelar ou desativar um arquivo *Query*, simplesmente digite o comando

SET FILTER TO ENTER

ou

USE

sem o nome do arquivo.

Para reativá-lo novamente, use o comando

SET FILTER TO FILE DEVEDOR

Veja na Figura 8.2 como deverá estar sua tela *Query* e seu arquivo gerado, a seguir.

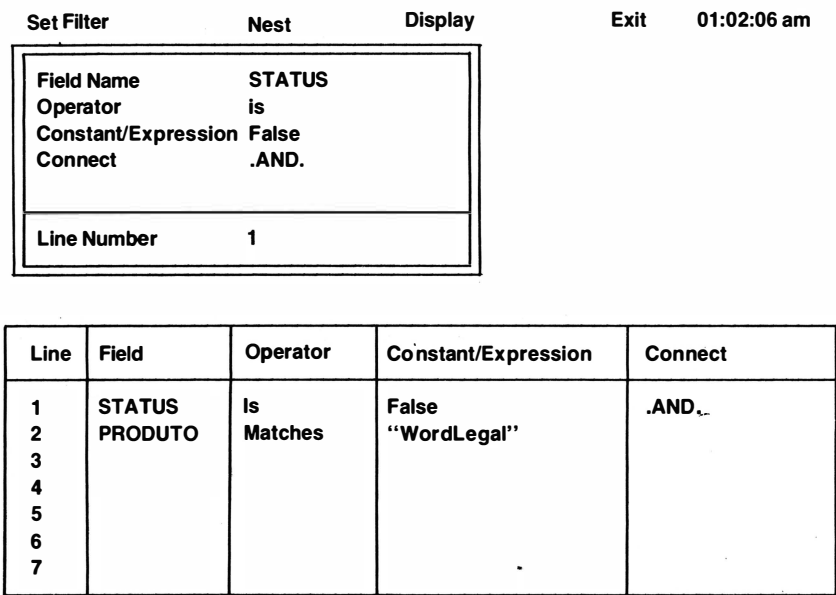


Figura 8.2: Tela *Query* parcial.

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
13	014	WordLegal	40	15.00	06/22/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.

Compare essa tela com as das páginas 102 e 105 e você verá que agora somente os registros 13, 16, 19 e 20 foram incluídos para atenderem às solicitações das duas condições. Lembre-se, você pode definir até 7 condições por cada arquivo.

Vamos continuar com nossos estudos de arquivos *Query*.

O Uso de Parênteses

Tal qual sua aplicação na Aritmética convencional, o uso de parênteses na confecção de arquivos *Query* faz com que o seu conteúdo seja executado primeiro. Por exemplo: a expressão $3 + 4 \div 2$ nos daria o resultado igual a 5 e não 3,5 que seria o resultado de $(3 + 4) \div 2$. O mesmo acontece com nossas informações quando usamos ou não os parênteses. E a coisa se complica mais, logicamente, à medida que usamos *mais de duas* condições. Vamos à prática!

Queremos acrescentar uma terceira condição ao nosso arquivo DEVEDOR. Além de ele exibir informações sobre todos os registros que corresponderem ao campo STATUS = .F., e ao campo PRODUTO = "WordLegal", queremos também que ele contenha todas as quantidades (QTDE) forem maiores (>) que ou igual a 40 unidades vendidas. Em outras palavras, a linha de comando terá duas alternativas:

STATUS = .F. .AND. PRODUTO = "WordLegal" .OR. QTDE > = 40

ou

STATUS = .F. .AND. (PRODUTO = "WordLegal" .OR. QTDE > = 40)

Na primeira alternativa, seu programa geraria todos os registros onde o campo STATUS fosse igual à condição .F. e onde o campo PRODUTO fosse igual a "WordLegal" ou a quantidade fosse maior ou igual a 40. Ou seja, essas condições seriam atendidas (não é o que queremos aqui):

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
1	001	LOTUS T.A.	40	4.00	07/20/88	.T.
2	002	WordLegal	60	15.00	07/20/88	.T.
6	010	ASMIBM PC	80	6.00	07/18/88	.T.
9	033	WordStar	80	4.00	07/08/88	.T.
10	012	MKT	80	4.00	06/30/88	.F.
12	008	MATFIN	80	4.00	06/22/88	.F.
13	014	WordLegal	40	15.00	06/22/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
18	001	dBASE III	80	6.00	06/08/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.

A segunda alternativa, a que desejamos *neste* exemplo traz a solução. O dBASE primeiro

processará o conteúdo dos parênteses e só *depois* de processá-lo é que fará a comparação com a condição .AND. .

Vamos usar a segunda opção do menu principal, *Nest*, para aprendermos a operar com parênteses. O submenu que aparece na tela é muito simples de ser analisado: *Add* é a parte do submenu que define a *entrada* (ou inserção) dos parênteses; *Remove* é a parte do submenu que define sua *saída* (ou remoção); as opções *Start* e *End* significam, respectivamente, as linhas onde os parênteses se iniciarão e as linhas onde eles terminarão.

Como exemplo, primeiro insira a terceira condição (linha 3) em seu arquivo *Query*, DEVEDOR de modo que, quando terminar, sua tela ampliada esteja como na Figura 8.3.

Line	Field	Operator	Constant/Expression	Connect
1	STATUS	Is	False	.AND.
2	PRODUTO	Matches	"WordLegal"	.OR.
3	QTDE	More than or equal	40	
4				
5				
6				
7				

Figura 8.3: Tela ampliada de *Query* (parcial).

A seguir, selecione a opção *Nest*, do menu principal e siga essas instruções:

MENU

Nest

Add

Start:2

End:3

Sua tela com os parênteses deverá estar como a Figura 8.4.

Agora salve seu trabalho com *Exit* e *Save*. Se quiser visualizar os novos resultados, chame os registros com *LIST* ou com o menu *Retrieve List*, como já fizemos anteriormente.

Sua tela com os registros do arquivo DEVEDOR do arquivo principal VENDAS deverá estar assim:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
10	012	MKT	80	4.00	06/30/88	.F.
12	008	MATFIN	80	4.00	06/22/88	.F.
13	014	WordLegal	40	15.00	06/22/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
18	001	dBASE III	80	6.00	06/08/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.

Set Filter

Nest

Display

Exit

01:33:13 am

Add

Start: 2

End: 3

Remove

Start: 0

End: 0

Line	Field	Operator	Constant/Expression	Connect
1	STATUS	Is	False	.AND.
2	(PRODUTO	Matches	"WordLegal"	.OR.
3	QTDE	More than or equal	40	)
4				
5				
6				
7				

Figura 8.4: Tela de *Query/Nest* com os parênteses.

Você também pode usar o comando *Display* do menu *Query* e as teclas PgDn e PgUp para visualizar todos os registros que preenchem as condições de busca especificadas. Este é o modo mais rápido de verificação do trabalho de “composição” dos arquivos *Query* e deve ser levado em consideração antes de salvarmos os arquivos, como precaução.

Testando Suas Habilidades

Vamos avaliar um pouco suas habilidades com arquivos *Query*. Vamos acrescentar mais uma condição .AND. ao arquivo. Só que desta vez não faremos o trabalho em conjunto, apenas sugeriremos as condições. Caberá a VOCÊ “construí-las”, desde a recuperação do arquivo até seu salvamento. Como parâmetro, além das condições, vamos exemplificar como sua tela *Query* e sua tela com os resultados do processamento deverão ficar após a conclusão do trabalho.

Sua tarefa será incluir a seguinte condição: “e” que o campo PRODUTO seja igual a “MKT”. A Sintaxe de seu comando completo deverá ficar assim:

STATUS = .F. .AND. ((PRODUTO = "WordLegal" .OR. QTDE > = "40") .AND. PRODUTO = "MKT")

Sua tela *Query* deverá ficar como na Figura 8.5.

Sua tela com os resultados finais deverá ficar assim:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
10	012	MKT	80	4.00	06/30/88	.F.

Line	Field	Operator	Constant/Expression	Connect
1	STATUS	Is	False	.AND.
2	((PRODUTO	Matches	"WordLegal"	.OR.
3	QTDE	More than or equal	40	.AND.
4	PRODUTO	Matches	"MKT"	)
5				
6				
7				

Figura 8.5: Resultado final da tela *Query*.

Note que agora, somente um registro atende a todas as condições especificadas!

Esperamos que você tenha conseguido chegar aos mesmos resultados (caso contrário alguma mensagem de erro como *"Invalid Filter"* deve ter sido exibida). Em todo caso, é sempre bom praticar com *Query* até que o domínio de suas técnicas esteja realmente consolidado.

No próximo capítulo trabalharemos com cálculos aritméticos, com datas e com variáveis de memória. No Capítulo 10, aprenderemos a usar o dBASE III Plus Relacional, criando e manipulando bancos de dados múltiplos e no Capítulo 11 abordaremos a criação de telas personalizadas.

Capítulo

9

Números, Datas e variáveis de Memória

Vamos chamar a estrutura de nosso último arquivo, VENDAS.DBF, usado no capítulo anterior para interações com *Query*. (Lembre-se, não mostramos sua estrutura no livro, apenas exibimos o arquivo pronto. Talvez você não tenha notado alguns detalhes importantes...)

Você pode chamar sua estrutura (do arquivo VENDAS.DBF *aberto*), com MODIFY STRUCTURE ou com DISPLAY STRUCTURE. Se quiser somente visualizar a estrutura do arquivo sem modificá-lo, use DISPLAY; caso queira modificá-lo, use MODIFY. De qualquer maneira, vamos usar um deles, DISPLAY, por exemplo, para que você veja como VENDAS.DBF *tem* que aparecer. Mais uma coisa, o comando MODIFY STRUCTURE pode ser chamado diretamente do *prompt*, com um arquivo ativo, ou via menu (*Modify – Database file*); já DISPLAY STRUCTURE tem de ser digitado diretamente do *prompt*.

Com VENDAS.DBF aberto, digite

DISPLAY STRUCTURE ENTER

e você verá a seguinte tela representando a estrutura completa de seu arquivo:

```
. display structure
Structure for database :B:vendas.dbf
Number of data records: 20
Date of last update   :04/02/88
```

Field	Field Name	Type	Width	Dec
1	CLIENTE	Character	4	
2	PRODUTO	Character	10	
3	QTDE	Numeric	3	
4	OTN	Numeric	6	2
5	VENC	Date	8	
6	STATUS	Logical	1	
** Total **			33	

Analisando a estrutura de nosso banco de dados VENDAS.DBF, vemos que, quando a preparamos (sem a sua intervenção), exibimos o resultado na página 102, especificamos alguns *tipos* de campo com dados numéricos e de datas, além de reservarmos um campo decimal com dois dígitos, para possíveis ocorrências de resultados com números decimais.

Caso você não tenha estruturado VENDAS.DBF assim (o que certamente ocorreu), faça-o agora. Use o comando MODIFY STRUCTURE diretamente do *prompt* ou *Modify-Database file*, via menu. Certifique-se de que seu arquivo fique exatamente igual à estrutura proposta.

Após ter editado a estrutura de VENDAS, chame o arquivo à tela de seu computador para que possamos executar algumas tarefas novas. Use os já conhecidos comandos LIST a partir do *prompt* ou via menu. Sua tela com o arquivo VENDAS.DBF, que usamos para criar nosso arquivo *Query*, DEVEDOR, no Capítulo 8, ainda deverá estar assim:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
1	001	LOTUS T.A.	40	4.00	07/20/88	.T.
2	002	WordLegal	60	15.00	07/20/88	.T.
3	003	WordLegal	20	15.00	07/20/88	.T.
4	003	dBASE III	15	6.00	07/20/88	.T.
5	001	APPLE	10	4.00	07/18/88	.T.
6	010	ASMIBM PC	80	6.00	07/18/88	.T.
7	042	ASMIBM PC	25	6.00	07/17/88	.T.
8	009	LOTUS T.A.	8	4.00	07/12/88	.T.
9	033	WordStar	80	4.00	07/08/88	.T.
10	012	MKT	80	4.00	06/30/88	.F.
11	005	MKT	10	4.00	06/30/88	.F.
12	008	MATFIN	80	4.00	06/22/88	.F.
13	014	WordLegal	40	15.00	06/22/88	.F.
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
17	037	ASMIBM PC	5	6.00	06/08/88	.F.
18	001	dBASE III	80	6.00	06/08/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.

SOMANDO CAMPOS

Primeiramente vamos somar, somente para efeito de treinamento, todo o campo OTN. Acione o comando

SUM OTN ENTER

diretamente do *prompt*, ou selecione, via menu

MENU

Retrieve

Sum

Construct a field list

OTN

->

Execute the command

O dBASE exibirá a seguinte mensagem, incluindo o resultado de nossa operação:

20 records summed

OTN

131.00

Você também pode somar, de uma só vez, *todos* os registros de seu arquivo. Simplesmente acione SUM sem especificar o campo. Por exemplo,

SUM ENTER

ou

```
MENU
Retrieve
Sum
Execute the command
N
```

exibirá o seguinte resultado:

```
      20 records summed
QTDE   OTN
  773   131.00
```

Pode-se também especificar somente alguns campos para serem somados. Em nosso caso específico só temos dois campos numéricos, mas se tivéssemos mais de dois, por exemplo, um campo hipotético chamado VALOR, poderíamos usar o comando

```
SUM QTDE, VALOR  ENTER
```

diretamente do *prompt*, ou

```
MENU
Retrieve
SUM
Construct a field list
QTDE
VALOR
->
Execute the command
```

via menu, para exibir somente os resultados da soma de todos os registros somente dos campos QTDE e VALOR, deixando o campo OTN de lado. Portanto, podemos somar todos os registros de um banco de dados com SUM, somente um registro com SUM *nome do campo*, ou mais de um registro (mas não necessariamente *todos*), com SUM *nome do campo*, *nome do campo*.

Agora vamos sofisticar mais um pouquinho nossa atividade. Somaremos somente registros que atendam a determinadas condições. Por exemplo, para somar o campo QTDE mais o campo OTN caso o código do campo CLIENTES seja 003, acione o comando

```
SUM QTDE, OTN FOR CLIENTE = '003'  ENTER
```

ou

```
MENU
Retrieve
```

Sum
Construct a field list
QTDE
OTN
Build a search condition
CLIENTE
 = *Equal To*
003
ENTER
No more conditions
Execute the command

e você terá o seguinte resultado em sua tela:

2 records summed	
QTDE	OTN
35	21.00

Chegaríamos ao mesmo resultado, via menu, se seleccionássemos:

MENU
Retrieve
Sum
Build a search condition
CLIENTE
 = *Equal To*
003
ENTER
No more conditions
Execute the command

Ou seja, não seria preciso especificar os campos QTDE e OTN porque somente esses dois campos são do tipo numérico e o comando SUM sem parâmetros, como já vimos antes, opera com todos os campos numéricos do arquivo.

Usando .AND. e .OR.

Você pode usar os comandos .AND. e .OR. para especificar mais de uma condição. Por exemplo, para especificar que queremos que o dBASE some todos os campos PRODUTO que tiverem o código WordLegal e todos os campos VENC que tiverem as datas igual a 07/20/88, digite

SUM FOR PRODUTO = 'WordLegal' .AND. VENC = CTOD('07/20/88') ENTER

ou (falaremos sobre CTOD() mais adiante, prefira o menu por enquanto)

MENU*Retrieve**Sum**Build a search condition***PRODUTO***= Equal To**WordLegal***ENTER***Combine with AND.***VENC***= Equal To**07/20/88**No more conditions**Execute the command*

Sua tela deverá exibir o seguinte resultado:

2 records summed	
QTDE	OTN
80	30.00

Note que com o comando SUM, apesar de indicarmos como parâmetros outros campos, o dBASE só relaciona, no resultado final, registros cujos campos sejam *numéricos*.

Poderíamos também usar *variáveis de memória* para efetuar cálculos *entre* campos. Por exemplo, se quiséssemos somar o conteúdo do campo QTDE com o conteúdo do campo OTN, teríamos as seguintes alternativas:

? 773+131 ENTER

(o sinal “?” significa “efetue a operação 773+131 e exiba o resultado”).

O resultado seria:

904

ou

SUM QTDE TO TOTAL ENTER

que somaria todos os registros do campo QTDE e os armazenaria em uma variável de memória chamada TOTAL; a seguir usaríamos o comando

SUM OTN TO FINAL ENTER

Ao fim das duas operações teríamos duas variáveis de memória chamadas TOTAL – que

conteria a soma de todos os registros do campo QTDE –, e FINAL – que conteria a soma de todos os registros do campo OTN. A seguir,

? TOTAL FINAL ENTER

nos daria a resposta

904

Para acharmos o valor correto das OTN's de todo nosso potencial de vendas do arquivo VENDAS.DBF, mude o sinal da operação:

? TOTAL * FINAL ENTER

e a resposta seria

101263.00

ou seja, nossa receita hipotética vendendo todos os produtos relacionados em nossa lista seria de 101.263 OTN's.

Para dividir os mesmos valores, simplesmente digite:

? 101263/131 ENTER

e o resultado será

773,00

Se digitarmos

? 101263/773 ENTER

nosso resultado será

131.00

e assim por diante. Lembre-se de que você também pode armazenar o produto 101.263 em uma variável chamada, por exemplo, MEMÓRIA, e executar as operações entre as três variáveis – TOTAL, FINAL e MEMÓRIA, que conteriam, respectivamente, os valores 773, 131 e 101.263.

Muito cuidado ao executar operações matemáticas com variáveis de *string* de caracteres alfabéticos (como as palavras TOTAL e FINAL, por exemplo). Se você tentar somar TOTAL + FINAL sem que essas duas palavras tenham sido previamente transformadas em variáveis, não obterá resultado algum. Como exemplo, tente somar as palavras QTDE +

OTN. Proceda da seguinte forma:

? QTDE + OTN ENTER

e você verá que o resultado será zero.

Porém, com o comando SUM (e não “?”) chegamos ao resultado esperado:

SUM QTDE + OTN ENTER

e o resultado seria

904

Para somar duas *palavras*, digite:

? “QTDE” + “OTN” ENTER

seu resultado será

QTDEOTN

O mesmo acontece com números. Digite

? ‘773’ + ‘131’ ENTER

e o resultado será a soma dos dois *strings*:

773131

Podemos também usar parênteses, como nossa operação com *Query*.

Digite

? (773 + 13.10) * 10 ENTER

O resultado será

7861.00

Ou seja, o dBASE primeiro processou os parênteses ($773 + 13.10 = 786.10$) e depois multiplicou seu resultado (786.10) ao número 10. Agora tente a mesma operação sem os parênteses. Digite

? 773 + 13.10 * 10 ENTER

e o resultado será

904.00

um resultado completamente diferente!

Pode-se também armazenar números em variáveis de memória, com o comando STORE ou seu equivalente “=”. Por exemplo, para executar a mesma operação acima, poderíamos simplesmente usar

STORE 773 TO TOTAL ENTER

ou

TOTAL = 773 ENTER

A seguir,

STORE 131 TO FINAL ENTER

ou

FINAL = 131 ENTER

A seguir, se solicitássemos ao dBASE

? TOTAL + FINAL ENTER

obteríamos o resultado esperado, ou seja

904

Podemos, ainda, efetuar cálculos com exponenciação, raiz quadrada, raiz cúbica, arredondamento, números inteiros, logaritmos e conversão de números negativos em positivos, além de soma, subtração, multiplicação, divisão e expressões aritméticas. Vejamos então alguns exemplos. Para elevar 12 ao quadrado:

? 12 ^ 2 ENTER

ou

? 12 ** 2 ENTER

Resultado na tela:

144.00

Para elevar 12 ao cubo:

```
? 12^3    ENTER
```

ou

```
? 12**3    ENTER
```

Resultado na tela:

```
1728.00
```

Usando SQRT

Para saber a raiz quadrada de 144 use o comando SQRT:

```
? SQRT(144)    ENTER
```

Resultado na tela:

```
12.00
```

ou, com o seguinte procedimento:

```
STORE 144 TO RAIZ    ENTER  
? SQRT(RAIZ)    ENTER
```

e o resultado será

```
12.00
```

Usando ABS

Para transformar um número negativo em positivo, use o comando ABS. Por exemplo:

```
? ABS(-144)    ENTER
```

exibirá o resultado

```
144
```

(não se esqueça de usar os parênteses!). Você também pode armazenar o número negativo em uma variável de *string* e depois convertê-la:

```
? STORE -144 TO A    ENTER  
? ABS(A)    ENTER
```

Resultado na tela:

144

Usando ROUND

Para arredondar um número, use a função ROUND com os parênteses:

VAR = 1.25879 ENTER

ou

STORE 1.25879 TO VAR ENTER

A seguir:

? ROUND(VAR,2) ENTER

e o resultado será

1.26000

Para três casas decimais:

? ROUND(VAR,3) ENTER

Resultado na tela:

1.25900

e assim por diante.

Arredonde para zero casas decimais e você obterá

1.00000

Mas há uma forma mais eficiente para se obter um número inteiro:

? INT(VAR) ENTER

fornece o resultado

1

ou seja, ela não arredonda nada, mas considera somente a parte inteira do número, eliminando *todas* as casas decimais.

Usando LOG e EXP

Pode-se também usar logaritmos e exponenciação com o dBASE, com as funções LOG e EXP, respectivamente. Tentemos alguns exemplos.

```
? LOG(5)    ENTER
```

e o programa responderá com

```
1.61
```

No caso oposto, isto é, com EXP:

```
? EXP(1.61)  ENTER
```

o programa responde com

```
5.00
```

Ou seja, o dBASE pode fornecer expoentes e logaritmos naturais de qualquer número desejado. (Entretanto, é importante ressaltar que os resultados nem sempre são demonstrados, em termos de estética, exatamente como você os vê nas páginas do livro). Por exemplo, no caso dos exemplos com EXP e LOG, sua tela deverá exibir um resultado semelhante a isto:

```
. ?LOG(5)  
1.61
```

ou isto:

```
. ?EXP(1.61)  
5.00
```

A quantidade de números decimais que porventura você deseje inserir em suas operações com logaritmos e exponenciação, por exemplo, pode ser programada por você mesmo ao definir o número. Simplesmente adicione quantos zeros forem necessários, após o ponto decimal, ao número entre parênteses (5.000000, 1.610000 etc.).

Usando AVERAGE

Podemos também trabalhar com a função AVERAGE, a partir do menu ou do *prompt*, para efetuarmos operações de cálculo das médias entre dois valores. Por exemplo, para tirar a média, de uma só vez, de *todos* os registros de seu arquivo, simplesmente acione AVERAGE sem especificar o campo:

AVERAGE ENTER

ou

MENU
Retrieve
Average
Execute the command

e você obterá o seguinte resultado:

20 records averaged (20 registros calculados)
QTDE OTN
39 6.55

Como acontece com SUM, também podemos trabalhar com campos específicos. Por exemplo:

AVERAGE QTDE ENTER

diretamente do *prompt*, ou

MENU
Retrieve
Average
Construct a field list
QTDE
 ->
Execute the command

via menu, para exibir somente os resultados da média de todos os registros do campo QTDE.

Usando COUNT

Vimos com AVERAGE que podemos aplicar a sintaxe de comandos de SUM para obter os mesmos resultados para uma média. Isto é, podemos usar AVERAGE e SUM da mesma forma, a partir do *prompt* ou do menu, para obtermos somas ou médias dos campos pré-determinados. Tente mais exemplos com AVERAGE, tal qual fizemos com SUM, até que você se sinta seguro e no controle absoluto de suas interações. Agora veremos como funciona o comando COUNT.

A função COUNT simplesmente conta o número de ocorrências em registros. Por exemplo, suponha que queiramos contar o número total de registros do arquivo VENDAS.DBF. Digite

COUNT ENTER

ou

MENU

Retrieve

Count

Execute the command

Sua resposta na tela seria:

20 records (20 registros)

indicando que temos um total de 20 registros alocados em nosso arquivo VENDAS.DBF, anteriormente usado para interações do tipo *Query* (lembra-se?). Agora vamos criar uma condição, ou especificação, para contarmos somente determinados campos pré-estabelecidos. Vamos supor uma contagem somente do campo PRODUTO das ocorrências de WordLegal. Digite

COUNT FOR PRODUTO = 'WordLegal' ENTER

a partir do *prompt*, ou

MENU

Retrieve

Count

Build a search condition

PRODUTO

= Equal To

WordLegal

No more conditions

Execute the command

a partir do menu.

Seu resultado será (espero...):

6 records (6 registros)

ou seja, o total desse produto (WordLegal) na coluna PRODUTO do banco de dados VENDAS.DBF.

OPERAÇÕES COM DATAS

São diversas as operações possíveis de se executar com datas, inclusive *aritmética* com

datas. Nesta seção introduziremos os conceitos e aplicações das funções DATE, RUN, TIME, CTOD, DTOC, CDOW e CMONTH.

Primeiramente, você deve saber que, para se trabalhar com datas no dBASE III Plus, é importante conhecer bem seus comandos, pois o programa não aceita comandos de datas a partir do menu. Isto quer dizer que você terá de usar comandos exclusivamente do *prompt* (não se esqueça de pressionar ESC para sair do *Assistant*). Lembre-se de que seu arquivo deve estar aberto para que você possa efetuar *qualquer* operação no dBASE.

Para ver a data “atual” do programa, digite

? DATE() ENTER

e para alterá-la para a data “real” (ou a data que você desejar),

RUN DATE ENTER

O dBASE mostrará a seguinte mensagem:

Current date is Tue 11-06-1986

Enter new date:_

(A data atual é Terça 11-06-1986

Insira a nova data)

Muita atenção, porém, porque o dBASE III Plus não rodará alguns comandos, como RUN, por exemplo, em determinadas classes de computadores “IBM - Compatíveis”. Nesses casos o melhor é testar o máximo possível seu programa junto ao fornecedor antes de se decidir pela aquisição de uma marca. Em tempo: caso você não tenha o programa COMMAND.COM em seu disquete de dBASE, digite o comando e *antes* de teclar ENTER, retire o disquete do dBASE III e coloque em seu lugar o disquete de seu DOS para que você receba a mensagem acima. Após inserir a data atual, tecle ENTER, retire o disquete do DOS e recoloque o disquete do sistema 2 do dBASE para continuar sua operação. A seguir digite

?DATE() ENTER

novamente, e veja o resultado.

Caso sua máquina esteja rodando de acordo, tudo o que você tem a fazer é digitar a data desejada para substituir a data anterior e, a seguir, pressionar ENTER para confirmá-la. Insira uma das datas de nosso arquivo VENDAS como exemplo, para que possamos trabalhar com ela nos próximos exercícios. Aceite como sugestão a data dos registros 1, 2, 3 e 4, que corresponde à data de 20-07-88. Após a alteração sua tela deverá estar assim:

Current date is Tue 11-06-1986

Enter new date: 07/20/88

Suponha que você queira listar todos os registros cujas datas de vencimento sejam iguais à data atual. Tudo o que você tem a fazer é abrir seu arquivo e digitar:

LIST FOR VENC = DATE() ENTER

e todos os registros correspondentes a essa data (registros 1, 2, 3 e 4) aparecerão em sua tela parcial.

Para visualizar uma data específica relativa a determinado campo, como por exemplo, a data do campo VENC igual a 07/20/88, encontramos uma operação um pouco menos simples (mas nem por isso difícil). Nesse caso tudo o que temos a fazer é, primeiro, *transformar* a data em dados do tipo caractere (*Character*) e depois efetuar a operação:

Primeiro armazenamos a data em uma variável de memória. Digite, como sugestão,

STORE '07/20/88' TO DATA ENTER

A seguir execute o restante da operação fazendo, ao mesmo tempo, a conversão do tipo do campo. Digite,

LIST FOR DTOC(VENC) = DATA ENTER

e sua tela deverá exibir todos os registros solicitados:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
1	001	LOTUS T.A.	40	4.00	07/20/88	.T.
2	002	WordLegal	60	15.00	07/20/88	.T.
3	003	WordLegal	20	15.00	07/20/88	.T.
4	003	dBASE III	15	6.00	07/20/88	.T.

O uso da função DTOC na realidade é muito simples. DTOC significa *Date TO Character*, ou seja, a conversão de datas para *strings* de caracteres.

Vejamos as funções relacionadas a operações com datas do dBASE III Plus na Tabela 9.1.

Você pode consultar a lista de funções relativas a operações com datas para gerar qualquer informação sobre uma data relacionada a um determinado campo e registro. Por exemplo, para visualizar todos os dias da semana do campo VENC, proceda da seguinte forma:

LIST VENC,CDOW(VENC) ENTER

e sua tela ficará assim:

Record #	venc	cdow (venc)
1	07/20/88	Wednesday
2	07/20/88	Wednesday
3	07/20/88	Wednesday

4	07/20/88	Wednesday
5	07/18/88	Monday
6	07/18/88	Monday
7	07/17/88	Sunday
8	07/12/88	Tuesday
9	07/08/88	Friday
10	06/30/88	Thursday
11	06/30/88	Thursday
12	06/22/88	Wednesday
13	06/22/88	Wednesday
14	06/18/88	Saturday
15	06/18/88	Saturday
16	06/12/88	Sunday
17	06/08/88	Wednesday
18	06/08/88	Wednesday
19	06/06/88	Monday
20	06/04/88	Saturday

FUNÇÃO	PROPÓSITO
CDOW	Dia da semana (por extenso)
CMONTH	Mês (por extenso)
CTOD	Conversão string-data
DAY	Dia do mês (dígito)
DOW	Dia da semana (domingo = 1, segunda = 2 etc.)
DTOC	Conversão data-string
MONTH	Mês (numérico)
TIME	Hora em HH:MM:SS
YEAR	Ano

Tabela 9.1: Funções de datas.

Este procedimento pode ser muito útil para agendar determinados compromissos como por exemplo saber com antecedência todas as datas de um pagamento ou recebimento. (Um detalhe importante deve ser notado: o dBASE não exibirá os dias no seu idioma, a não ser, é claro, que este seja o Inglês – o idioma natural do programa!)

Pode-se também solicitar ao dBASE que mostre somente datas que caiam no sábado, para um melhor controle de recebimentos, aplicações etc.:

LIST FOR CDOW(VENC) = 'Saturday' ENTER

nos mostrará uma tela com todos os vencimentos do campo VENC que porventura forem acontecer num sábado:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.

Você também pode exibir o mês de determinado vencimento. Por exemplo, para exibir o mês em que os vencimentos do campo VENC caem, por extenso (já que os respectivos meses já se encontram no local da planilha na forma original, ou seja, expressa em dígitos), digite

LIST VENC,CMONTH(VENC) ENTER

e sua tela deverá estar como a seguir:

Record #	venc	cmonth(venc)
1	07/20/88	July
2	07/20/88	July
3	07/20/88	July
4	07/20/88	July
5	07/18/88	July
6	07/18/88	July
7	07/17/88	July
8	07/12/88	July
9	07/08/88	July
10	06/30/88	June
11	06/30/88	June
12	06/22/88	June
13	06/22/88	June
14	06/18/88	June
15	06/18/88	June
16	06/12/88	June
17	06/08/88	June
18	06/08/88	June
19	06/06/88	June
20	06/04/88	June

Note que geramos apenas o mês relativo ao campo VENC, digitando o nome do campo (VENC) seguinte a LIST e incluindo-o entre os parênteses da sintaxe da função de conversão CMONTH. Caso quisesse, você poderia não indicar o nome do campo junto a LIST e o programa geraria uma tela com todos os campos, mas atenderia a nossa solicitação, incluindo os nomes dos respectivos meses. Experimente!

Podemos gerar também uma tela contendo todas as datas que caíam somente em determinado mês. Por exemplo, digite

LIST VENC FOR CMONTH(VENC) = 'June' ENTER

neste caso sua tela ficaria assim:

Record #	venc
10	06/30/88
11	06/30/88
12	06/22/88
13	06/22/88
14	06/18/88

15	06/18/88
16	06/12/88
17	06/03/88
18	06/08/88
19	06/06/88
20	06/04/88

ou, se você preferisse,

LIST FOR CMONTH(VENC) = 'June' ENTER

e sua tela ficaria completa, com os resultados desejados:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
10	012	MKT	80	4.00	06/30/88	.F.
11	005	MKT	10	4.00	06/30/88	.F.
12	008	MATFIN	80	4.00	06/22/88	.F.
13	014	WordLegal	40	15.00	06/22/88	.F.
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
17	037	ASMIBM PC	5	6.00	06/08/88	.F.
18	001	dBASE III	80	6.00	06/08/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.

Bem, vamos agora ao caso clássico de Aritmética com datas. Vamos supor que você queira alterar, em 30 dias, o vencimento de suas duplicatas ou compromissos. A alteração será para adiante, portanto, partirá das datas atuais para as novas datas.

Primeiro vamos obter uma visão de nosso novo arquivo. Para simplesmente visualizar essa nova tela e ver como ficariam os futuros vencimentos, simplesmente digite

LIST VENC + 30 ENTER

e sua tela deverá aparecer assim:

Record #	venc + 30
1	08/19/88
2	08/19/88
3	08/19/88
4	08/19/88
5	08/17/88
6	08/17/88
7	08/16/88
8	08/11/88
9	08/07/88
10	07/30/88
11	07/30/88
12	07/22/88
13	07/22/88

14	07/18/88
15	07/18/88
16	07/12/88
17	07/08/88
18	07/08/88
19	07/06/88
20	07/04/88

Caso você tenha interesse em mantê-la com as alterações, poderá editá-la com *Edit* ou *Browse*.

Você também pode especificar a data com a qual será comparado o prazo esgotado de 30 dias. Por exemplo, suponha que queiramos visualizar os prazos vencidos a 30 dias a partir de nossa data de pagamento de 07/20/88. Digite:

LIST FOR CTOD('07/20/88') – VENC >= 30 .ENTER

e sua tela ficará assim:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
17	037	ASMIBM PC	5	6.00	06/08/88	.F.
18	001	dBASE III	80	6.00	06/08/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.

Não se esqueça de usar as funções de conversão ou seu programa não lhe dará o resultado esperado (porém lhe dará algum resultado, mesmo que este não signifique coisa alguma para você).

Para saber a diferença entre duas datas, simplesmente subtraia a data da outra. Se desejar números negativos, subtraia uma data maior de uma data menor e vice-versa quando desejar o resultado positivo. Por exemplo, para saber a quantos dias estamos num determinado projeto, que hipoteticamente tenha se iniciado em 12 de julho de 1984 até hoje (hipoteticamente – 09 de maio de 1988), proceda da seguinte forma:

LIST CTOD('05/09/88') – CTOD('07/12/84') ENTER

e o resultado será:

Record #	ctod('05/09/88')-ctod('07/12/84')
1	1397
2	1397
3	1397
4	1397
5	1397
6	1397

7	1397
8	1397
9	1397
10	1397
11	1397
12	1397
13	1397
14	1397
15	1397
16	1397
17	1397
18	1397
19	1397
20	1397

Ao recebimento de qualquer mensagem de erro, verifique se você não deixou inadvertidamente de usar a programação da função de conversão apropriada. Não deixe também de usar aspas ou apóstrofes para definir a data.

Indexação com Datas

Indexar bancos de dados com operações de datas é muito semelhante à indexação já utilizada anteriormente em nosso banco de dados CLIENTES.DBF, na fase de classificação de bancos de dados. Por exemplo, se desejarmos visualizar uma listagem ou tela com todos os campos baseados na ordem cronológica inversa (VENC), tudo o que temos a fazer é especificar uma data qualquer *maior* que as datas incluídas em seu programa. Ou seja:

INDEX ON CTOD('10/30/90') – VENC TO B:NOVOVENC ENTER

Usando seus conhecimentos para listar (LIST) seu banco de dados, chame sua tela e veja o resultado:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
1	001	LOTUS T.A.	40	4.00	07/20/88	.T.
2	002	WordLegal	60	15.00	07/20/88	.T.
3	003	WordLegal	20	15.00	07/20/88	.T.
4	003	dBASE III	15	6.00	07/20/88	.T.
5	001	APPLE	10	4.00	07/18/88	.T.
6	010	ASMIBM PC	80	6.00	07/18/88	.T.
7	042	ASMIBM PC	25	6.00	07/17/88	.T.
8	009	LOTUS T.A.	8	4.00	07/12/88	.T.
9	033	WordStar	80	4.00	07/08/88	.T.
10	012	MKT	80	4.00	06/30/88	.F.
11	005	MKT	10	4.00	06/30/88	.F.
12	008	MATFIN	80	4.00	06/22/88	.F.
13	014	WordLegal	40	15.00	06/22/88	.F.
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
17	037	ASMIBM PC	5	6.00	06/08/88	.F.

18	001	dBASE III	80	6.00	06/08/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.

Da mesma forma, se você quiser classificar seu banco de dados normalmente, sem exigências especiais, digite:

INDEX ON VENC TO OUTRO ENTER

Após a LISTagem de seu banco de dados você verá a seguinte tela:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
20	019	WordLegal	60	6.00	06/04/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
17	037	ASMIBM PC	5	6.00	06/08/88	.F.
18	001	dBASE III	80	6.00	06/08/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
12	008	MATFIN	80	4.00	06/22/88	.F.
13	014	WordLegal	40	15.00	06/22/88	.F.
10	012	MKT	80	4.00	06/30/88	.F.
11	005	MKT	10	4.00	06/30/88	.F.
9	033	WordStar	80	4.00	07/08/88	.T.
8	009	LOTUS T.A.	8	4.00	07/12/88	.T.
7	042	ASMIBM PC	25	6.00	07/17/88	.T.
5	001	APPLE	10	4.00	07/18/88	.T.
6	010	ASMIBM PC	80	6.00	07/18/88	.T.
1	001	LOTUS T.A.	40	4.00	07/20/88	.T.
2	002	WordLegal	60	15.00	07/20/88	.T.
3	003	WordLegal	20	15.00	07/20/88	.T.
4	003	dBASE III	15	6.00	07/20/88	.T.

Você também pode indexar mais de um campo, ou seja, combinar dois campos para sua classificação, sendo um deles o campo relativo às datas (VENC), por exemplo.

Para classificar o banco de dados VENDAS.DBF pelo campo VENC combinado com o campo PRODUTO, sugerimos:

INDEX ON PRODUTO + DTOC(VENC) TO RESULTADO ENTER

Agora acione o comando

LIST ENTER

e você deverá ver o seguinte resultado na tela:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
5	001	APPLE	10	4.00	07/18/88	.T.
17	037	ASMIBM PC	5	6.00	06/08/88	.F.

7	042	ASMIBM PC	25	6.00	07/17/88	.T.
6	010	ASMIBM PC	80	6.00	07/18/88	.T.
8	009	LOTUS T.A.	8	4.00	07/12/88	.T.
1	001	LOTUS T.A.	40	4.00	07/20/88	.T.
12	008	MATFIN	80	4.00	06/22/88	.F.
10	012	MKT	80	4.00	06/30/88	.F.
11	005	MKT	10	4.00	06/30/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
13	014	WordLegal	40	15.00	06/22/88	.F.
2	002	WordLegal	60	15.00	07/20/88	.T.
3	003	WordLegal	20	15.00	07/20/88	.T.
9	033	WordStar	80	4.00	07/08/88	.T.
18	001	dBASE III	80	6.00	06/08/88	.F.
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
4	003	dBASE III	15	6.00	07/20/88	.T.

Se digitarmos a mesma linha *sem* o argumento “DTC(VENC)”:

INDEX ON PRODUTO TO RESULTADO ENTER

sua tela ficará assim:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
5	001	APPLE	10	4.00	07/18/88	.T.
6	010	ASMIBM PC	80	6.00	07/18/88	.T.
7	042	ASMIBM PC	25	6.00	07/17/88	.T.
17	037	ASMIBM PC	5	6.00	06/08/88	.F.
1	001	LOTUS T.A.	40	4.00	07/20/88	.T.
8	009	LOTUS T.A.	8	4.00	07/12/88	.T.
12	008	MATFIN	80	4.00	06/22/88	.F.
10	012	MKT	80	4.00	06/30/88	.F.
11	005	MKT	10	4.00	06/30/88	.F.
2	002	WordLegal	60	15.00	07/20/88	.T.
3	003	wordLegal	20	15.00	07/20/88	.T.
13	014	WordLegal	40	15.00	06/22/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.
9	033	WordStar	80	4.00	07/08/88	.T.
4	003	dBASE III	15	6.00	07/20/88	.T.
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
18	001	dBASE III	80	6.00	06/08/88	.F.

Como vimos, as operações com datas são semelhantes em quase todos os aspectos com exceção, é claro, das aplicações das funções especiais de conversão de datas para caracteres, meses, dias da semana etc.

Bem, vejamos agora a hora do programa. Digite

? TIME() ENTER

e o programa exibirá a hora do sistema no formato HH:MM:SS

18:23:09

Para modificar a hora e inserir sua hora atual, proceda da mesma forma que fizemos com DATE. Digite o comando

RUN TIME ENTER

e simplesmente insira a hora desejada

Current time is 18:23:18

Enter new time:_

no espaço deixado com o cursor.

Usando TOTAL

Vamos retomar um pouco nossa discussão com arquivos indexados. Você pode achar estranho ter de voltar a esse assunto novamente, mas fizemos isso de propósito, para que você possa aproveitar essa revisão compulsória.

A diferença é que agora usaremos um novo comando, TOTAL, para somar os dados de um determinado campo. TOTAL funciona como uma variável de memória no sentido de que ele também solicita a criação de um arquivo paralelo tal qual acontece com as operações comuns de INDEX. Por exemplo, com seu arquivo VENDAS ativo, digite

INDEX ON CLIENTE TO B:VENDAS ENTER

A seguir podemos usar o comando TOTAL que nos mostrará os totais *por cliente* de todo o nosso banco de dados, resumindo-os em registros únicos para cada campo, considerando todos os campos. Digite:

TOTAL ON CLIENTE TO RESUMO ENTER

Veja o que a tela diz:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
1	001	LOTUS T.A.	40	4.00	07/20/88	.T.
5	001	APPLE	10	4.00	07/18/88	.T.
18	001	dBASE III	80	6.00	06/08/88	.F.
2	002	WordLegal	60	15.00	07/20/88	.T.
3	003	WordLegal	20	15.00	07/20/88	.T.
4	003	dBASE III	15	6.00	07/20/88	.T.

11	005	MKT	10	4.00	06/30/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
12	008	MATFIN	80	4.00	06/22/88	.F.
8	009	LOTUS T.A.	8	4.00	07/12/88	.T.
6	010	ASMIBM PC	80	6.00	07/18/88	.T.
10	012	MKT	80	4.00	06/30/88	.F.
13	014	WordLegal	40	15.00	06/22/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
9	033	WordStar	80	4.00	07/08/88	.T.
17	037	ASMIBM PC	5	6.00	06/08/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
7	042	ASMIBM PC	25	6.00	07/17/88	.T.

Agora, chame seu novo arquivo, RESUMO, e compare o resultado:

USE RESUMO ENTER
LIST ENTER

(Lembre-se de que você ainda pode chamar o menu *Assist* para chamar um arquivo, com *Set Up*, *Retrieve* etc., mas nesse caso isso seria um pouco demorado, já que estamos trabalhando exclusivamente com o *prompt* neste estágio.)

Observe agora que seu arquivo RESUMO, na realidade um subarquivo do arquivo principal VENDAS, exibe somente os totais de cada cliente, unificados em um só registro para cada cliente. Foi o que pedimos!

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
1	001	LOTUS T.A.	130	14.00	07/20/88	.T.
2	002	WordLegal	60	15.00	07/20/88	.T.
3	003	WordLegal	35	21.00	07/20/88	.T.
4	005	MKT	10	4.00	06/30/88	.F.
5	007	WordLegal	20	4.00	06/06/88	.F.
6	008	MATFIN	80	4.00	06/22/88	.F.
7	009	LOTUS T.A.	8	4.00	07/12/88	.T.
8	010	ASMIBM PC	80	6.00	07/18/88	.T.
9	012	MKT	80	4.00	06/30/88	.F.
10	014	WordLegal	40	15.00	06/22/88	.F.
11	019	WordLegal	60	6.00	06/04/88	.F.
12	021	dBASE III	20	6.00	06/18/88	.F.
13	022	dBASE III	10	6.00	06/18/88	.F.
14	033	WordStar	80	4.00	07/08/88	.T.
15	037	ASMIBM PC	5	6.00	06/08/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
17	042	ASMIBM PC	25	6.00	07/17/88	.T.

As Variáveis Públicas e Privadas

Vamos dar maior velocidade às nossas interações. Acione o comando

DISPLAY MEMORY ENTER

A seguir veja o resultado na tela.

```

0 variables defined,      0 bytes used
256 variables available, 6000 bytes available

```

O comando que acabamos de digitar nos levou a uma tela que diz: “não há nenhuma variável *definida* até o momento e, portanto, também não há nenhum *byte* utilizado. O que temos são 256 variáveis disponíveis e 6000 *bytes* disponíveis para essas mesmas variáveis.”

Isto quer dizer que ainda não utilizamos nenhuma variável (definida por nós, os usuários) *pública* (PUBLIC). Quer dizer também, talvez, que já podemos ter usado (como realmente fizemos!) variáveis que não definimos como variáveis públicas, no início da operação, e por isso mesmo elas desapareceram quando saímos do programa principal. Em outras palavras, as variáveis *públicas* são diferentes das variáveis *privadas* no que se refere à sua permanência ou não no programa, *permanentemente*. No dBASE, por exemplo, todas as variáveis são, por *default*, variáveis privadas, isto é, variáveis que somente são consideradas durante sua interação com elas. Vamos a um exemplo prático (você já deve ter percebido que o nosso “forte” é aliar a teoria à prática – e ambas à indispensável qualidade!).

Iniciação à Programação

Vamos criar o pequeno programa a seguir com uma variável definida como pública e depois analisar uma nova solicitação de DISPLAY MEMORY. Digite:

PUBLIC SNAME ENTER

Acabamos de definir nossa variável SNAME como pública. Agora vamos fazer um pequeno programa para podermos usá-la a seguir. Digite:

ACCEPT 'Digite seu.sobrenome:' TO SNAME

Imediatamente após a mensagem que acabamos de digitar, vemos a pergunta “Digite seu sobrenome” na tela. Digite, em resposta à solicitação do computador, o seu sobrenome. A seguir digite novamente o comando

DISPLAY MEMORY ENTER

e analise o resultado de sua tela agora e veja que ela apresenta uma pequena alteração. Agora podemos ver que há uma variável denominada *pub* (de *pública*), do tipo caractere (“C”), cujo preenchimento foi feito com a resposta à solicitação do programa, com o sobrenome do autor deste livro, Dalton, e que toda a operação consumiu 8 *bytes* tendo como referência uma variável definida (SNAME).

A seguir, crie outra variável, NOME, para uma solicitação do primeiro nome do autor. Digite:

PUBLIC NOME ENTER

e, a seguir,

ACCEPT 'Digite seu nome:' TO NOME

Responda com o nome do autor, Roger.

Agora, se digitar **DISPLAY MEMORY** novamente, você verá que temos duas variáveis definidas pelo usuário (você) com um total de 15 *bytes*. Não se esqueça também de observar que a mensagem na parte inferior da tela diz que só temos 254 variáveis disponíveis para definirmos e um total de 5985 *bytes* para essa tarefa.

```
SNOME   pub   C  "Dalton"
NOME     pub   C  "Roger"
      2 variables defined,      15 bytes used
      254 variables available,  5985 bytes available
```

Uma importante observação tem de ser feita. Quando definimos nossa variável nos dois exemplos anteriores, fomos direto à própria definição, deixando de lado qualquer outra preparação. Isto quer dizer que poderíamos ter evitado que as solicitações para inserção do nome e do sobrenome do autor aparecessem ao mesmo tempo, juntamente com a própria digitação do comando **ACCEPT**. Como exemplo, vamos separar uma área na memória especialmente para podermos rodar nosso programa quando quisermos, com o comando **DO**. Para isso, temos de usar a instrução **MODIFY COMMAND** da seguinte forma: Primeiro criaremos um arquivo (para ser executado posteriormente, quando desejarmos) chamado, por exemplo, de **GAVETA**. A seguir, digitaremos o mesmo programa que digitamos anteriormente (a definição das duas variáveis públicas) e finalmente armazenaremos o programa para ser posteriormente executado com o comando **DO**. Digite:

MODIFY COMMAND B:GAVETA ENTER

A seguir digite:

PUBLIC NOME ENTER

seguido do comando **ACCEPT** exatamente como o exemplo anterior. Ao terminar, salve o seu trabalho com **Ctrl-W**. Depois de gravado, execute seu programa

DO GAVETA ENTER

e o programa só acabará quando você digitar seu nome.

Agora, se digitarmos

? NOME ENTER

a tela exibirá o último nome inserido.

No capítulo sobre programação falaremos mais sobre variáveis, pois elas são indispensáveis a esse tópico.

No próximo Capítulo trataremos do dBASE III Plus Relacional e a criação e manipulação de bancos de dados múltiplos. No Capítulo 11, trataremos da impressão de relatórios simples e múltiplos e no Capítulo 12 introduziremos a programação com o dBASE III Plus.

Capítulo

10

**dBASE III Plus Relacional
Criação e Manipulação de
Bancos de Dados Múltiplos**

Neste Capítulo trataremos da criação e da manipulação de bancos de dados múltiplos, com o dBASE relacional. Aqui, realmente estaremos vendo o que é o chamado de Gerenciamento de Arquivos ou de Banco de dados.

Com o dBASE III Plus, pode-se combinar e inter-relacionar mais de um arquivo *sem* ter de sair do programa para o sistema operacional. Pode-se também transferir dados de um arquivo para outro de forma parcial e total.

Há muitas razões para se querer, por exemplo, dividir um banco de dados em dois pequenos (ou menores), ao invés de utilizar um arquivo maior, mais confuso (truncado) e, conseqüentemente, mais lento, durante sua confecção. Tomemos como exemplo nosso banco de dados VENDAS.DBF. Vamos dividi-lo em dois arquivos mais ágeis para que possamos trabalhar com eles de acordo com nossa real necessidade no futuro. Chame seu arquivo VENDAS que está no acionador B com *Set Up e Database file*, ou a partir do *prompt* com USE e o nome do arquivo.

Agora vamos dividir Vendas em dois arquivos (na maioria das publicações sobre dBASE III Plus, trata-se somente da *criação* de dois arquivos para posterior combinação. Aqui nós iremos *dissecar* esse potencial de combinação de arquivos, dividindo nosso arquivo VENDAS.DBF em dois e, posteriormente, juntando-os por meio de diversos comandos do *prompt* e do menu). Primeiro, com o arquivo VENDAS.DBF aberto, acione o comando:

COPY TO GRANA FIELDS OTN, VENC, STATUS ENTER

ou, usando os comandos equivalentes do menu (o autor deste livro, durante a fase de elaboração e testes, preferiu um maior uso dos comandos do *prompt*, por serem eles mais rápidos à medida que sua criatividade/domínio da linguagem aumenta):

MENU

Organize

Copy

B:

Construct a field list

GRANA.DBF

OTN

VENC

STATUS

—>

Execute the command

Agora abra seu novo arquivo GRANA.DBF com *Set Up, Database file, B:*, ou com o comando USE do *prompt* e, a seguir, liste-o com *Retrieve List, Execute the command, N*, ou simplesmente com o comando LIST, a partir do *prompt*. Sua tela deverá estar assim:

Record #	CLIENTE	OTN	VENC	STATUS
1	001	4.00	07/20/88	.T.

2	002	15.00	07/20/88	.T.
3	003	15.00	07/20/88	.T.
4	003	6.00	07/20/88	.T.
5	001	4.00	07/18/88	.T.
6	010	6.00	07/18/88	.T.
7	042	6.00	07/17/88	.T.
8	009	4.00	07/12/88	.T.
9	033	4.00	07/08/88	.T.
10	012	4.00	06/30/88	.F.
11	005	4.00	06/30/88	.F.
12	008	4.00	06/22/88	.F.
13	014	15.00	06/22/88	.F.
14	021	6.00	06/18/88	.F.
15	022	6.00	06/18/88	.F.
16	038	6.00	06/12/88	.F.
17	037	6.00	06/08/88	.F.
18	001	6.00	06/08/88	.F.
19	007	4.00	06/06/88	.F.
20	019	6.00	06/04/88	.F.

Nosso próximo caminho é indexar o arquivo GRANA.DBF pelo *campo chave* CLIENTE (pode-se usar qualquer campo chave – desde que ele seja representado nos dois arquivos que serão combinados, com mesmo tipo, tamanho e nome). Acione o comando:

INDEX ON CLIENTE TO GRANA

(ATENÇÃO: Em algumas publicações pede-se para colocar o nome do acionador. Não faça isto agora, ou seu programa não considerará o arquivo GRANA e sim B:GRANA – que não será o nome utilizado durante nossas interações!)

A seguir vamos repetir toda a operação que acabamos de fazer com VENDAS.DBF para criar um outro arquivo, chamado PERFIL.DBF. Proceda da mesma forma para copiar somente os campos pré-definidos. Acione o comando:

COPY TO PERFIL FIELDS CLIENTE, PRODUTO, QTDE ENTER

ou seu equivalente do menu

MENU

Organize

Copy

B:

Construct a field list

PERFIL.DBF

CLIENTE

PRODUTO

QTDE

->

Execute the command

Agora abra seu novo arquivo PERFIL.DBF e, a seguir, como fez com GRANA.DBF, liste-o e veja sua nova tela.

Record #	CLIENTE	PRODUTO	QTDE
1	001	LOTUS T.A.	40
2	002	WordLegal	60
3	003	WordLegal	20
4	003	dBASE III	15
5	001	APPLE	10
6	010	ASMIBM PC	80
7	042	ASMIBM PC	25
8	009	LOTUS T.A.	8
9	033	WordStar	80
10	012	MKT	80
11	005	MKT	10
12	008	MATFIN	80
13	014	WordLegal	40
14	021	dBASE III	20
15	022	dBASE III	10
16	038	WordLegal	30
17	037	ASMIBM PC	5
18	001	dBASE III	80
19	007	WordLegal	20
20	019	WordLegal	60

Seus dois arquivos GRANA e PERFIL são na verdade o antigo arquivo VENDAS.DBF dividido em dois. Até aqui tudo bem, mas por que razão fizemos isto? Bem, como já dissemos no início do capítulo nosso objetivo é, na realidade, conectar dois arquivos e não separá-los! Então vamos ao prato principal. Antes porém, vamos indexar (lembre-se: arquivos combinados têm de ser indexados!) nosso novo arquivo PERFIL.DBF. Acione o comando:

INDEX ON CLIENTE TO PERFIL ENTER

ou use o menu *Organize* para proceder de forma idêntica à que fizemos com o arquivo GRANA, com exceção do nome do arquivo, é claro.

A seguir, acione o comando LIST *antes* de abrir PERFIL. Compare esta operação com a operação anterior, quando listamos GRANA *após* tê-lo aberto com USE (ou com o menu). Repare que, quando não abrimos um arquivo logo após sua indexação e em seguida o listamos, o que vemos é a tela do arquivo indexado e não a tela do arquivo original:

Record #	CLIENTE	PRODUTO	QTDE
1	001	LOTUS T.A.	40
5	001	APPLE	10
18	001	dBASE III	80
2	002	WordLegal	60
3	003	WordLegal	20
4	003	dBASE III	15
11	005	MKT	10
19	007	WordLegal	20

12	008	MATFIN	80
8	009	LOTUS T.A.	8
6	010	ASMIBM PC	80
10	012	MKT	80
13	014	WordLegal	40
20	019	WordLegal	60
14	021	dBASE III	20
15	022	dBASE III	10
9	033	WordStar	80
17	037	ASMIBM PC	5
16	038	WordLegal	30
7	042	ASMIBM PC	25

Querendo visualizar a tela de PERFIL.DBF original, simplesmente abra o arquivo PERFIL e use o comando LIST (ou *List*, do menu).

Podemos perceber que, em ambos os arquivos, somente 20 campos foram indexados porque cada um dos arquivos só tem 20 campos – inclusive o arquivo original, VENDAS.DBF. Um outro detalhe importante é a opção *Tools* do menu principal. Essa opção permite também a cópia de arquivos, porém é um comando limitado, pois só permite a cópia de arquivos *inteiros* e, portanto, não nos serve no momento, para nossa prática de seleção parcial de VENDAS.DBF.

Usando APPEND FROM

Um comando muito simples para se usar em combinações de arquivos é o comando APPEND FROM, diretamente do *prompt*. Com este comando pode-se apenas copiar – e conectar – campos de um mesmo tipo (definidos na estruturação dos arquivos), considerando-se também o tamanho do arquivo receptor em relação ao arquivo fonte, o arquivo original. Como um exemplo bastante simples, vamos copiar, da mesma forma que fizemos com GRANA.DBF e com PERFIL.DBF – a partir de VENDAS.DBF –, um novo arquivo. Como prática, vamos usar o próprio arquivo VENDAS e, baseando-se nas informações que acabamos de receber, copie todo o arquivo VENDAS para um novo arquivo chamado PRATICA.DBF. Use a opção *Tools* do menu principal para exercitar sua funcionalidade. Acione o menu da seguinte forma (primeiro feche todos os arquivos com CLEAR ALL – que fecha todos os arquivos e as variáveis de memória –, ou com CLOSE DATABASES – que fecha todos os arquivos exceto as variáveis de memória):

```
MENU
Tools
Copy file
B:
VENDAS.DBF
B:
PRATICA.DBF
```

ou, se você preferir, acione a mesma instrução a partir do *prompt*:

COPY FILE VENDAS.DBF TO PRATICA.DBF

e os resultados serão idênticos, ou seja, abra primeiro seu novo arquivo, PRATICA.DBF, e liste-o. Aproveite e abra o arquivo VENDAS e liste-o também (mas não ao mesmo tempo). Se você os comparar, verá que eles são idênticos em tudo, exceto no que se refere ao nome. Veja a tela de PRATICA.DBF:

Record #	CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
1	001	LOTUS T.A.	40	4.00	07/20/88	.T.
2	002	WordLegal	60	15.00	07/20/88	.T.
3	003	WordLegal	20	15.00	07/20/88	.T.
4	003	dBASE III	15	6.00	07/20/88	.T.
5	001	APPLE	10	4.00	07/18/88	.T.
6	010	ASMIBM PC	80	6.00	07/18/88	.T.
7	042	ASMIBM PC	25	6.00	07/17/88	.T.
8	009	LOTUS T.A.	8	4.00	07/12/88	.T.
9	033	WordStar	80	4.00	07/08/88	.T.
10	012	MKT	80	4.00	06/30/88	.F.
11	005	MKT	10	4.00	06/30/88	.F.
12	008	MATFIN	80	4.00	06/22/88	.F.
13	014	WordLegal	40	15.00	06/22/88	.F.
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
17	037	ASMIBM PC	5	6.00	06/08/88	.F.
18	001	dBASE III	80	6.00	06/08/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.

Agora vamos copiar um campo do arquivo PRATICA em um novo pequeno arquivo chamado ATRASADO.DBF. Acione o comando:

COPY TO ATRASADO FIELD PRODUTO FOR PRODUTO = 'WordLegal' ENTER

Ainda no *prompt*, para não perdermos mais tempo, digite:

USE ATRASADO ENTER

Vejamos como ficou nosso mais novo arquivo, ATRASADO.DBF. Acione o comando LIST ou *List* do menu **Retrieve**:

Record #	PRODUTO
1	WordLegal
2	WordLegal
3	WordLegal
4	WordLegal
5	WordLegal
6	WordLegal

Podemos ver que nosso arquivo ATRASADO.DBF contém somente o campo que definimos previamente, PRODUTO, e comportando somente as condições solicitadas, ou seja,

vemos somente os registro dos aplicativos WordLegal. Mas há uma coisa que você pode não estar entendendo. Por quê gerar um arquivo como este? Boa pergunta. E a resposta é simples: exclusivamente para que você tenha mesmo que *pensar*! Na realidade este arquivo (ATRASADO.DBF) somente funcionaria bem se tivéssemos solicitado o campo do CLIENTE em atraso juntamente com o campo PRODUTO, para que a identificação ficasse completa. Compare sua tela de ATRASADO.DBF com a tela que acabamos de propor, mentalmente, ou gere um outro arquivo ATRASADO.DBF se preferir (se for essa a sua escolha, tecla "Y" à pergunta sobre se você deseja sobregravar o antigo arquivo ATRASADO).

Aqui, não faz muita diferença qual dos dois arquivos você escolheu. Para nós o importante será a próxima interação com o programa, onde veremos como o comando APPEND FROM funciona.

Bem, agora que já temos um arquivo exclusivo para praticar (PRATICA.DBF), vamos acionar APPEND FROM, a partir do *prompt*, para finalmente conectá-lo ao arquivo ATRASADO.DBF. Primeiro abra o arquivo receptor, que neste caso será o arquivo PRATICA. Acione o comando:

USE PRATICA ENTER

A seguir acione o comando APPEND FROM:

APPEND FROM ATRASADO

Em seguida liste PRATICA.DBF e veja o resultado.

4	003	dBASE III	15	6.00	07/20/88	.T.
5	001	APPLE	10	4.00	07/18/88	.T.
6	010	ASMIBM PC	80	6.00	07/18/88	.T.
7	042	ASMIBM PC	25	6.00	07/17/88	.T.
8	009	LOTUS T.A.	8	4.00	07/12/88	.T.
9	033	WordStar	80	4.00	07/08/88	.T.
10	012	MKT	80	4.00	06/30/88	.T.
11	005	MKT	10	4.00	06/30/88	.F.
12	008	MATFIN	80	4.00	06/22/88	.F.
13	014	WordLegal	40	15.00	06/22/88	.F.
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
17	037	ASMIBM PC	5	6.00	06/08/88	.F.
18	001	dBASE III	80	6.00	06/08/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.
21		WordLegal			/ /	.F.
22		WordLegal			/ /	.F.
23		WordLegal			/ /	.F.
24		WordLegal			/ /	.F.
25		WordLegal			/ /	.F.
26		WordLegal			/ /	.F.

Na realidade o resultado, considerando-se o arquivo ATRASADO.DBF que escolhemos, não nos serve para nada. Sugerimos que você crie um arquivo do tipo ATRASADO, porém com registros próprios, *diferentes* dos registros do arquivo receptor. Isto não seria mais lógico (advinhe por quê)?

Bem, já aprendemos como usar o comando APPEND FROM (e como *não* usá-lo). Agora vamos passar para um tipo de exercício mais complicado: a combinação de dois arquivos diferentes, com campos diferentes, numa tela só.

Usando JOIN WITH

Antes de terminar este Capítulo estaremos dominando três tipos de comando – dois a partir do *prompt* e um a partir do menu – de conexão de arquivos. O primeiro deles será o comando JOIN WITH, diretamente do *prompt*.

Para usar JOIN WITH, ou seja, para trabalhar com dois arquivos simultaneamente, primeiro temos que selecionar, com SELECT, duas áreas distintas de memória ativa. Essas áreas podem ser nomeadas até 10 vezes, de 1 a 10 ou de A a J. Por exemplo, para trabalharmos com os arquivos PERFIL.DBF e GRANA.DBF e, conseqüentemente, para juntarmos os dois num só arquivo novamente, temos de acionar (pressione ENTER após cada linha):

CLEAR ALL	(fecha todos os arquivos abertos)
SELECT 1	(seleciona a área de trabalho 1)
USE GRANA INDEX GRANA	(abre GRANA.DBF e GRANA.NDX)
SELECT 2	(seleciona a área de trabalho 2)
USE PERFIL INDEX PERFIL	(abre PERFIL.DBF e PERFIL.NDX)

e a seguir:

```
JOIN WITH GRANA TO NOVOARQ FOR CLIENTE = GRANA -> CLIENTE FIELDS
GRANA -> CLIENTE, GRANA -> OTN, GRANA -> VENC, GRANA -> STATUS,
PERFIL -> PRODUTO, PERFIL-> QTDE    ENTER
```

Analisando os Procedimentos

Vamos pesquisar o (um tanto quanto extenso) comando acima. Primeiro, executamos os procedimentos de seleção das áreas de trabalho e abertura dos arquivos a serem trabalhados – que foram auto-explicados. A seguir, digitamos o comando JOIN WITH que diz: “conecte o arquivo PERFIL.DBF, o último selecionado com SELECT, ao arquivo GRANA.DBF e crie, ao juntar os dois, um novo arquivo chamado NOVOARQ que atenda à condição especificada (o campo CLIENTE *nos dois* arquivos de trabalho), gerando do arquivo GRANA somente os campos solicitados (nesse caso, OTN, VENC e STATUS – nesta ordem) e do arquivo PERFIL também os campos solicitados (PRODUTO e QTDE)”.

Repare que, para definir a condição, usamos *FOR condição* (o campo chave comum aos dois) = *nome do arquivo* (nesse caso GRANA.DBF) e para definirmos os campos – e sua ordem de apresentação – dos dois arquivos de trabalho, usamos a opção *FIELDS* seguida do nome do arquivo gerador dos campos e da seta (feita no teclado com o hífen e o símbolo de maior que (>)).

Agora, vejamos o resultado desse trabalho. Talvez você já tenha notado que, após teclar ENTER, o programa tenha mostrado a mensagem “28 records joined” que diz que 28 registros foram acrescentados ao novo arquivo NOVOARQ.DBF e não 20, como seria de esperar. Vamos listar o arquivo para apreciá-lo melhor. Primeiro, vamos fechar todos os arquivos novamente. Se não fizermos isso, não poderemos, por exemplo, listar o arquivo GRANA.DBF diretamente, e nem abri-lo com USE – pois ele já está aberto (porém está alocado à área de trabalho 1), mas poderemos listar imediatamente o novo arquivo recém-gerado, NOVOARQ.DBF ou abrir e gerar PERFIL.DBF que foi o arquivo em cuja área de trabalho (2) operamos por último. Por exemplo, para visualizar o resultado de sua operação de união dos arquivos GRANA.DBF e PERFIL.DBF, acione agora o comando LIST.

LIST ENTER

e veja a tela que aparece!

Record #	CLIENTE	PRODUTO	QTDE
1	001	LOTUS T.A.	40
2	002	WordLegal	60
3	003	WordLegal	20
4	003	dBASE III	15
5	001	APPLE	10
6	010	ASMIBM PC	80
7	042	ASMIBM PC	25
8	009	LOTUS T.A.	8
9	033	WordStar	80
10	012	MKT	80
11	005	MKT	10
12	008	MATFIN	80
13	014	WordLegal	40
14	021	dBASE III	20
15	022	dBASE III	10
16	038	WordLegal	30
17	037	ASMIBM PC	5
18	001	dBASE III	80
19	007	WordLegal	20
20	019	WordLegal	60

Repare que esta não é a tela de NOVOARQ e sim a tela de PERFIL.DBF, o último arquivo aberto (com SELECT 2). Para ver o conteúdo de NOVOARQ.DBF, primeiro abra-o com o comando USE (ou com o menu, se preferir – mas com o qual não sugerimos nenhuma operação no momento, pois continuaremos algum tempo com o *prompt*). A seguir, digite o comando

LIST ENTER

e veja (finalmente!) o resultado de nosso extenso comando JOIN WITH:

Record #	CLIENTE	OTN	VENC	STATUS	PRODUTO	QTDE
1	001	4.00	07/20/88	.T.	LOTUS T.A.	40
2	001	4.00	07/18/88	.T.	LOTUS T.A.	40
3	001	6.00	06/08/88	.F.	LOTUS T.A.	40
4	001	4.00	07/20/88	.T.	APPLE	10
5	001	4.00	07/18/88	.T.	APPLE	10
6	001	6.00	06/08/88	.F.	APPLE	10
7	001	4.00	07/20/88	.T.	dBASE III	80
8	001	4.00	07/18/88	.T.	dBASE III	80
9	001	6.00	06/08/88	.F.	dBASE III	80
10	002	15.00	07/20/88	.T.	WordLegal	60
11	003	15.00	07/20/88	.T.	WordLegal	20
12	003	6.00	07/20/88	.T.	WordLegal	20
13	003	15.00	07/20/88	.T.	dBASE III	15
14	003	6.00	07/20/88	.T.	dBASE III	15
15	005	4.00	06/30/88	.F.	MKT	10
16	007	4.00	06/06/88	.F.	WordLegal	20
17	008	4.00	06/22/88	.F.	MATFIN	80
18	009	4.00	07/12/88	.T.	LOTUS T.A.	8
19	010	6.00	07/18/88	.T.	ASMIBM PC	80
20	012	4.00	06/30/88	.F.	MKT	80
21	014	15.00	06/22/88	.F.	WordLegal	40
22	019	6.00	06/04/88	.F.	WordLegal	60
23	021	6.00	06/18/88	.F.	dBASE III	20
24	022	6.00	06/18/88	.F.	dBASE III	10
25	033	4.00	07/08/88	.T.	WordStar	80
26	037	6.00	06/08/88	.F.	ASMIBM PC	5
27	038	6.00	06/12/88	.F.	WordLegal	30
28	042	6.00	07/17/88	.T.	ASMIBM PC	25

Agora podemos observar que alguma coisa não deu certo. Como podemos ter 28 registros se os dois arquivos usados para o processo de combinação só têm 20? A resposta é muito simples. O comando que acabamos de usar só funciona bem com arquivos cujo campo chave (CLIENTE nesse caso) *não* seja repetido! Ou seja, com o comando JOIN WITH que usamos a pouco, não podemos conectar com perfeição os arquivos GRANA e PERFIL por terem eles, no seu campo chave CLIENTE, *registros repetidos* (001 e 003). Assim, os resultados foram parciais (o arquivo *foi* conectado), mas inúteis. Conclusão: devemos usar o comando acima, no mesmo formato usado, com arquivos que não repitam registros em seu campo chave!

Para listar PERFIL.DBF na tela, abra-o com USE e depois acione o comando LIST. Tente abrir GRANA.DBF. Acione o comando USE GRANA e veja o resultado na tela.

```
. USE GRANA
File is already open.
?
USE GRANA
Dou you want some help? (Y/N)
```

A mensagem do programa diz que o arquivo já está aberto e pergunta se você deseja al-

guma ajuda. Tecle “N” e deixe que Roger Dalton lhe ajude.

As alternativas que temos para poder abrir e listar GRANA.DBF são duas: 1) pode-se usar o comando **SELECT 1** novamente (que é onde o arquivo se encontra desde o início deste mesmo comando) e sua área de trabalho estará acessada – e o arquivo aberto. O passo seguinte é listar o arquivo; 2) pode-se usar o comando **CLEAR ALL** para fechar todos os arquivos (inclusive as áreas do tipo **SELECT**) e a seguir acionar os comandos **USE GRANA** e **LIST** (ou **F3** no lugar de **LIST**) nesta ordem.

Em ambos os casos, ou seja, após **SELECT 1** ou **CLEAR ALL**, todos os procedimentos para abertura de arquivos voltam ao normal – nenhum arquivo continua vinculado ao comando **SELECT**.

Usando SET RELATION TO

Uma outra modalidade de comando usada para conectar arquivos diferentes é o comando **SET RELATION TO *campo chave* INTO *nome do arquivo***. Por exemplo, para executarmos a mesma operação anterior (feita com **JOIN WITH**), e somente os 20 registros corretos serem incluídos, podemos acionar os seguintes comandos (compare com a operação anterior):

CLOSE DATABASES	(fecha todos os arquivos)
SELECT A	(seleciona a área de trabalho A)
USE GRANA INDEX GRANA	(abre GRANA.DBF e GRANA.NDX)
SELECT B	(seleciona a área de trabalho B)
USE PERFIL INDEX PERFIL	(abre PERFIL.DBF e PERFIL.NDX)

Agora que já definimos as áreas de trabalho como A e B, vamos criar a *relação* entre o campo chave e o último arquivo aberto. Acione o comando:

SET RELATION TO CLIENTE INTO PERFIL ENTER

Selecione uma das áreas contendo um dos programas como, por exemplo, a área A:

SELECT A

Eis aqui seu novo comando:

LIST CLIENTE, B -> PRODUTO, B -> QTDE, OTN, VENC, STATUS ENTER

e sua tela deverá ficar assim:

Record #	cliente	b->produto	b->qtde	otn	venc	status
1	001	LOTUS T.A.	40	4.00	07/20/88	.T.
5	001	LOTUS T.A.	40	4.00	07/18/88	.T.
18	001	LOTUS T.A.	40	6.00	06/08/88	.F.

2	002	WordLegal	60	15.00	07/20/88	.T.
3	003	WordLegal	20	15.00	07/20/88	.T.
4	003	WordLegal	20	6.00	07/20/88	.T.
11	005	MKT	10	4.00	06/30/88	.F.
19	007	WordLegal	20	4.00	06/06/88	.F.
12	008	MATFIN	80	4.00	06/22/88	.F.
8	009	LOTUS T.A.	8	4.00	07/12/88	.T.
6	010	ASMIBM PC	80	6.00	07/18/88	.T.
10	012	MKT	80	4.00	06/30/88	.F.
13	014	WordLegal	40	15.00	06/22/88	.F.
20	019	WordLegal	60	6.00	06/04/88	.F.
14	021	dBASE III	20	6.00	06/18/88	.F.
15	022	dBASE III	10	6.00	06/18/88	.F.
9	033	WordStar	80	4.00	07/08/88	.T.
17	037	ASMIBM PC	5	6.00	06/08/88	.F.
16	038	WordLegal	30	6.00	06/12/88	.F.
7	042	ASMIBM PC	25	6.00	07/17/88	.T.

Analizando os Procedimentos

O comando que acabamos de utilizar – e que gerou a tela dos dois arquivos conectados – tem algumas diferenças fundamentais em relação ao comando JOIN WITH anterior. Uma delas é o próprio nome do arquivo que é inexistente. (No caso anterior denominamos o arquivo de NOVOARQ.DBF, lembra-se?) A outra diferença fundamental é, obviamente, sua capacidade de exibir o novo arquivo corretamente (e diretamente na tela via LIST), somente com o número de registros contidos nos dois arquivos originais, ou seja, 20. Observe também sua característica prática de não precisarmos repetir o nome do arquivo que precede a seta (→) e só precisarmos indicar a área correspondente à área que *não* foi selecionada por último (ou seja, se precedermos o comando com SELECT A, devemos indicar com a letra da outra área – nesse caso B –, e com a seta, os campos referentes a este arquivo. Não é necessário usar o nome do arquivo em uso – GRANA.DBF, o arquivo selecionado por último –, ou sua letra correspondente – nesse caso, A – e as setas correspondentes – com o comando JOIN anterior isso também é possível). Pode-se também ignorar (como foi feito com SET RELATION, nos dois comandos – JOIN e SET RELATION –, a definição do arquivo em uso, ou seja, se usarmos B→ para definirmos as relações entre o arquivo e o campo, não precisaremos usar, na definição dos campos do outro arquivo, a letra A e o símbolo → (composto pelo hífen e o sinal de maior que). Muito complicado? Então vejamos um exemplo. Consulte o exemplo *modificado* de acordo com a explicação que acabamos de dar e compare-o com os dois comandos que aprendemos neste capítulo para conectar dois arquivos – JOIN e SET RELATION:

Comando JOIN modificado:

```
JOIN WITH A TO NOVOARQ FOR CLIENTE = A -> CLIENTE FIELDS A -> CLIENTE,
A -> OTN, A -> VENC, A -> STATUS, PRODUTO, QTDE      ENTER
```

Mais um detalhe importante. Podemos substituir, no momento da definição das áreas de trabalho com SELECT, as letras A e B pelos números 1 e 2 (lembre-se de que pode-se

usar até 10 definições, que vão de A a J e de 1 a 10), porém, muita atenção: com os números *não* podemos deixar de colocar os nomes dos arquivos, ou seja, não podemos substituir A -> OTN por 1 -> OTN, por exemplo.

Vejamos agora nosso terceiro comando, ou seja, o que podemos fazer com relação à conexão desses mesmos dois arquivos via menu.

Usando View

Para começar, *View* é um comando que pertence ao menu, como dissemos antes, portanto, vamos tratar de passar para o menu com F2 ou digitando ASSIST diretamente no *prompt*. (Não é preciso usar letras maiúsculas. Aliás isso não é preciso com nenhum comando do dBASE nem para nomes de campo – somente, que isto fique claro, os nomes dos *registros* contidos em cada campo é que devem ser digitados, quando dentro de algum comando, exatamente iguais, para que as condições estabelecidas no comando sejam atendidas de acordo.)

Agora, no menu *Assistant*, vamos criar um arquivo para receber as cópias de GRANA.DBF e de PERFIL.DBF tal qual fizemos com SET RELATION. Vamos criar um arquivo receptor chamado VNOVO. Destaque a opção *Create* do menu principal e a seguir a subopção *View*, assim:

```
MENU
Create
View
B:
ENTER
GRANA.DBF
GRANA.NDX
<-
PERFIL.DBF
PERFIL.NDX
->
->
```

Analisando os Procedimentos

Primeiro criamos um novo arquivo (ainda vazio) chamado VNOVO (de *View* + NOVO) que vem a ser nosso arquivo receptor (o equivalente de NOVOARQ que criamos com o comando JOIN WITH). Em seguida, selecionamos, na mesma ordem que fizemos com SELECT, os arquivos GRANA.DBF e GRANA.NDX – seu equivalente indexado. Continuando, após pressionar a seta à esquerda para voltar ao submenu de arquivos .DBF, selecionamos da mesma forma os arquivos PERFIL.DBF e PERFIL.NDX e saímos do menu

Create com o pressionamento da tecla com a seta à direita duas vezes, até a opção ***Relate***, onde termina nossa primeira análise.

Agora, no menu ***Relate***, vamos à nossa segunda operação. Selecione as seguintes subopções do menu:

```
MENU
Relate
GRANA.DBF
PERFIL.DBF
F10
CLIENTE
ENTER
->
->
```

Analizando os Procedimentos

Nosso segundo menu nos leva ao relacionamento, propriamente dito, entre os dois arquivos e seu campo chave, CLIENTE. Primeiro selecionamos nosso já conhecido arquivo GRANA.DBF seguido da seleção do não menos conhecido arquivo PERFIL.DBF. Em seguida, pressionamos F10, o que nos leva diretamente ao submenu de campos onde optamos pelo campo CLIENTE, o *pivô* de nossa operação. Fechamos esta etapa pressionando ENTER e, em seguida, pressionando duas vezes a tecla com a seta à direita para pararmos no início da opção *Set Fields*.

Para continuar, vamos ao nosso terceiro menu. Selecione:

```
MENU
Set Fields
GRANA.DBF
->
PERFIL.DBF
CLIENTE
->
->
->
```

Analizando os Procedimentos

Nesta terceira e penúltima etapa, selecionamos os campos que desejamos visualizar na tela, com GRANA.DBF seguido da tecla com a seta à direita para considerar todos os campos que nos foram apresentados pelo submenu (se quiséssemos poderíamos ter indicado somente alguns campos com ENTER sobre os campos desejados). Em seguida, selecionamos o arquivo PERFIL.DBF seguido da seleção (na realidade uma *eliminação*) do campo

CLIENTE. Para finalizar pressionamos a tecla com a seta à direita três vezes até chegarmos à opção **Exit** do menu principal, nossa última parada.

No menu **Exit** selecionamos a opção *Save* e nosso arquivo VNOVO já estará pronto para ser aberto e apresentado na tela.

Abra seu novo arquivo com *Set Up, View, B:*, VNOVO.VUE e visualize-o com *Retrieve, List, Execute the command, N* (ou “Y”, caso prefira gerar sua listagem diretamente na impressora, além de na tela). A seguir veja sua tela do menu *View*:

Record #	CLIENTE	OTN	VENC	STATUS	PRODUTO	QTDE
1	001	4.00	07/20/88	.T.	LOTUS T.A.	40
5	001	4.00	07/18/88	.T.	LOTUS T.A.	40
18	001	6.00	06/08/88	.F.	LOTUS T.A.	40
2	002	15.00	07/20/88	.T.	WordLegal	60
3	003	15.00	07/20/88	.T.	WordLegal	20
4	003	6.00	07/20/88	.T.	WordLegal	20
11	005	4.00	06/30/88	.F.	MKT	10
19	007	4.00	06/06/88	.F.	WordLegal	20
12	008	4.00	06/22/88	.F.	MATFIN	80
8	009	4.00	07/12/88	.T.	LOTUS T.A.	8
6	010	6.00	07/18/88	.T.	ASMIBM PC	80
10	012	4.00	06/30/88	.F.	MKT	80
13	014	15.00	06/22/88	.F.	WordLegal	40
20	019	6.00	06/04/88	.F.	WordLegal	60
14	021	6.00	06/18/88	.F.	dBASE III	20
15	022	6.00	06/18/88	.F.	dBASE III	10
9	033	4.00	07/08/88	.T.	WordStar	80
17	037	6.00	06/08/88	.F.	ASMIBM PC	5
16	038	6.00	06/12/88	.F.	WordLegal	30
7	042	6.00	07/17/88	.T.	ASMIBM PC	25

No próximo capítulo introduziremos a programação – inclusive com macros e arquivos em lote. No Capítulo 12, finalmente trataremos da impressão de relatórios simples e múltiplos e no Capítulo 13, falaremos de projeto e desenvolvimento de programas, incluindo técnicas de depuração.

Capítulo

11

**Programação com o
dBASE III Plus: Macros
e Decisão Automatizada
(O “SuperMacro” auto@)**

INTRODUÇÃO À PROGRAMAÇÃO

Neste capítulo trataremos de introduzir – e complementar – os conceitos e a prática da programação com o dBASE III Plus. Iniciaremos nossos estudos com um pequeno programa prático – comentado – e, a partir daí progrediremos, sempre comentando suas diversas fases e alternativas, até a construção de um pequeno “complexo” de programas que poderão ser interligados – ou não, dependendo da vontade do usuário –, cujo programa de controle, batizado sugestivamente de *auto@*, poderá controlar dois outros programas utilitários: um programa de controle de clientes – de nosso já conhecido arquivo *CLIENTES.DBF* – e um programa de emissão de etiquetas de endereçamento – que funcionará com um novo arquivo especialmente criado para essa tarefa. Ao término de todas as sessões, você poderá usar cada programa individualmente ou conectados entre si via programa de controle geral, o futuramente popular **auto@.prg**. Todo esse pequeno complexo foi denominado pelo autor de **SuperMacro**, para que você, o usuário, pudesse entender melhor as capacidades de programação com arquivos de comando (programas), macros e tomada de decisão automatizada e, posteriormente, dar seus próprios nomes a seus próprios programas.

O Primeiro Arquivo de Comandos

Vamos então ao *prompt*. Este é um dos detalhes importantes para se iniciar na programação, já que não se pode programar com o menu!

Para começar nosso primeiro (pequeno) programa, trabalharemos com **MODIFY COMMAND**, já visto anteriormente no Capítulo 10. Digite então:

```
MODIFY COMMAND REF      ENTER
```

para abrir uma tela especial de programação transformada imediatamente em um arquivo chamado *Ref* (de referência).

Escolhemos o nome Referência (*ref.prg*) porque queremos lembrar-lhe da necessidade de se ter um arquivo de referência paralelo para operar com esses programas, mas você pode; se preferir, usar qualquer outro nome mais sugestivo para você, desde que ele atenda às normas de nomeação de arquivos (máximo de 8 caracteres mais o ponto e três caracteres de extensão, *sem* sinais de pontuação ou espaços).

Agora, com a tela de **MODIFY COMMAND** ativa, copie *exatamente* como no Programa 11.1 (por enquanto):

```
? trim(nome), snome
? tel
```

Programa 11.1: Início de programação.

Analizando os Procedimentos

Vemos, em nossa primeira tela de programação, um menu específico de MODIFY COMMAND. Em poucos instantes falaremos sobre ela.

O comando “?” indica que queremos exibir alguma coisa na tela (ou na impressora). O comando TRIM (do verbo *aparar* em inglês) indica que queremos o campo NOME, que está entre parênteses para indicar que ele será o primeiro campo considerado e (“ , ”) deve ser exibido lado-a-lado com o campo SNOME *aparando* os espaços que normalmente aparecem quando chamamos um arquivo com LIST. Em seguida vemos mais uma linha de comando que diz para o programa exibir logo abaixo do nome, o campo TEL. Vejamos agora a prática. Digite, para salvar seu trabalho no disco, Ctrl-W. A seguir digite:

USE CLIENTES ENTER

e ao *prompt* digite:

DO REF

Sua tela deverá exibir o nome de nosso primeiro cliente do arquivo CLIENTES.DBF criado no início de nossas interações com o livro e que agora, após tantos exercícios, só possui três campos: SNOME, NOME e TEL.

Vamos continuar com nossa progressão. Copie agora mais uma parte do programa para novos testes e comentários. Digite:

MODIFY COMMAND REF ENTER

para entrar novamente no modo de programação com o programa REF.PRG. A seguir, copie as outras três últimas linhas do programa exatamente como você as vê no Programa 11.2. Agora vamos copiar a primeira linha. Para isso, coloque o cursor na posição do ponto de interrogação, entre no modo Insert pressionando a tecla INS e tecle ENTER. Todo o texto do programa cairá uma linha para baixo e você poderá então digitar a linha que passa agora a ser a primeira linha do programa.

```
do while .not. eof( )
? trim(nome),sname
? tel
skip
enddo
return
```

Programa 11.2: REF.PRG parcial.

Digite:

CTRL-W

para sair da tela de MODIFY COMMAND e gravar o programa REF.PRG. Novamente digite:

USE CLIENTES ENTER

e

DO REF ENTER

Observe que agora seus resultados foram um pouco melhores. Dessa vez todo o arquivo CLIENTES foi gerado e nós saberemos porque no próximo parágrafo. Antes porém, veja o resultado de seu último comando:

Record No.	1
Celso DeLanteuil	
(021)067-7074	
Record No.	2
Anna NARCISI	
(021)645-2525	
Record No.	3
Helena C. Narcisi	
Record No.	4
Gisella Narcisi	
Record No.	5
Joao Pedro	
Record No.	6
Joca Pedro	
Record No.	7
Antonio Pedroso	
Record No.	8
Larissa Pinduba	
Record No.	9
Lilica Pinduba	
Record No.	10
Jose Silva	

Analisando os Procedimentos

Agora a primeira linha de nosso programa diz: “execute o programa (DO) enquanto (WHILE) ele não (.NOT.) chegar até o fim do arquivo (EOF())”. Já sabemos tudo sobre as próximas duas linhas. A quarta linha, SKIP, significa “salte a próxima instrução” e é complementada por ENDDO que será a instrução saltada até que sua condição seja atendi-

da, ou seja, até que todos os registros sejam exibidos. A próxima instrução (RETURN) retorna para o início do programa se ele *não* atingir o fim, ou seja, até que o programa acabe de exibir todos os registros. E é por isso que todos os registros foram exibidos!

Repare que seu arquivo foi de fato todo impresso, mas seus números de registro também o foram juntamente com "Record No.". Para evitar que isto aconteça, podemos digitar os comandos:

USE CLIENTES ENTER

(para abrir CLIENTES.DBF),

SET TALK OFF ENTER

(para desativar a exibição dos registros)

e

DO REF ENTER

(para executar o programa REF.PRG).

Note que agora os registros que compõem o arquivo CLIENTES.DBF aparecem novamente sem as identificações de registro.

Separando os Espaços

Agora vamos separar os espaços entre os registros, para que no futuro eles possam ser exibidos e/ou impressos mais esteticamente. Chame seu programa REF.PRG novamente com MODIFY COMMAND REF e, ao entrar na tela do programa, copie as três linhas com pontos de interrogação que faltam, como na ilustração do Programa 11.3.

```
do while .not. eof( )
? trim(nome), snome
? tel
?
?
skip
enddo
return
```

Programa 11.3: REF.PRG mais estético.

Novamente salve seu programa com Ctrl-W e retorne ao *prompt*. Repita os procedimentos de abertura de CLIENTES e em seguida acione o comando:

DO REF ENTER

novamente, para ver o resultado de mais esta etapa.

**Celso DeLanteuil
(021)067-7074**

**Anna NARCISI
(021)645-2525**

Helena C. Narcisi

Gisella Narcisi

•
•
•
(etc.)

Agora podemos ver que os registros aparecem com mais espaço entre eles (exatamente três linhas).

Vejamos agora o que acontece quando digitamos em seguida

DO REF ENTER

sem abrir o arquivo CLIENTES.DBF:

Na realidade nada acontece porque já chegamos ao fim do arquivo, como bem programamos. Mas e se desejarmos gerar mais uma vez nossas resumidas (por enquanto) “etiquetas”? Bem vamos explicar isso na prática.

Retorne à tela de ref.prg via MODIFY COMMAND REF e copie as instruções que faltam de acordo com o Programa 11.4.

******* Endereçamento *******

```
go top
do while .not. eof( )
    ? trim(nome),snome
    ? tel
    ?
    ?
    ?
    skip
enddo
return
```

Programa 11.4: REF.PRG completo.

***** Endereçamento *****

```
go top
do while .not. eof( )
    ? trim(nome),some
    ? tel
    ?
    ?
    ?
    skip
enddo
return
```

Programa 11.4: REF.PRG completo.

Note que também inserimos uma pequena modificação, além de um título para identificarmos melhor o programa. Copie tudo como você vê (mais abaixo faremos os comentários). Quando acabar de copiar, salve tudo com Ctrl-W, abra CLIENTES.DBF e execute DO REF. Agora já não precisamos de abrir CLIENTES toda vez que quisermos rodar nosso programa. Experimente!

Analizando os Procedimentos

Com ENTER na primeira letra da primeira linha, abaixamos o texto para inserir o título após os asteriscos. Esses asteriscos (equivalentes da instrução REM da linguagem BASIC) servem como mensagens somente e nunca são impressos, ou seja, tudo o que digitarmos após um ou mais asteriscos não aparecerá no resultado do programa. Colocamos também a linha de comando Go Top que indica ao programa para ir automaticamente para o início do arquivo quando terminar o processo de leitura. Assim, sempre que você quiser chamar REF, é só digitar DO REF e o programa começará do início automaticamente, sem necessidade de se abrir o arquivo principal a cada vez (caso o arquivo esteja em uso). Note também que alteramos a estética do programa, para mostrar ao leitor/usuário o local do programa onde ocorre o *loop*. (Esta questão da estética lógica dependerá de cada usuário.)

Saia da tela de MODIFY COMMAND para o *prompt* e digite DO REF com o arquivo CLIENTES ativo. Repita a operação e veja que o programa roda novamente e assim sucessivamente até que você pare de comandar.

O MENU DE PROGRAMAÇÃO DE MODIFY COMMAND

Edit:B:ref.prg

CURSOR < - - - > Char: ← → Word: Home End Line: ^ ← ^ → Reformat: ^ KB	UP DOWN Line: ↑ ↓ Page: PgUp PgDn Find: ^ KF Refind: ^ KL	DELETE Char: Del Word: ^ T Line: ^ Y	Insert Mode: Ins Insert Line: ^ N Save: ^ W Abort: Esc Read file: ^ KR Write file: ^ KW
---	--	--	--

Figura 11.1: Menu de MODIFY COMMAND.

O menu de programação de MODIFY COMMAND (Figura 11.1) é muito parecido com os menus de EDIT e APPEND, porém, há necessidade de alguns comentários adicionais: na coluna CURSOR, podemos encontrar a instrução *Reformat: ^KB*. Esta instrução indica que, tal qual um processador de textos, pode-se reformatar um “parágrafo” ou uma linha de comando, quando fazemos inserções no texto (mais adiante mostraremos alguns exemplos práticos dessa especificidade). Na coluna UP DOWN, vemos o comando *Find ^KF* e *Refind ^KL* que querem dizer “encontre a palavra X” e “reencontre a palavra X”, ou seja, sua sintaxe é: Ctrl-KF *string* e Ctrl-KL *mesmo string*. Já conhecemos bem o modo DELETE, então vamos à coluna Insert Mode onde vemos *Read file, (^KR)* que copia tudo o que estiver exatamente acima de seu cursor, na própria posição do cursor e *Write file*, que grava o programa, ou texto sendo elaborado no disquete, sem sair do modo MODIFY COMMAND (como ocorre com Ctrl-W). O restante das instruções você já deve conhecer (senão conhece, elas são auto-explicativas).

MACROS: PROGRAMANDO ETIQUETAS DE ENDEREÇAMENTO

Agora passemos a um programa um pouco mais avançado. Um *macro* é um programa com condições específicas embutidas, contendo o símbolo “&”. Nesta seção, faremos um programa do tipo macro e começaremos a usar a impressora para imprimir nossas etiquetas de endereçamento. Tudo isso será feito com um arquivo especial chamado LIVRARIA.DBF que criaremos agora especificamente para nossas operações com o programa ETIQUETA.PRG, que será construído posteriormente com MODIFY COMMAND, da mesma forma que construímos o programa REF.PRG.

O Arquivo de Endereçamento

Primeiro estruture seu banco de dados LIVRARIA.DBF como no exemplo seguinte (copie exatamente como na Figura 11.2 que segue, inclusive o campo Memo, que será usado posteriormente para outras tarefas não menos importantes).

Structure for database: B:livraria.dbf
 Number of data records: 16
 Date of last update: 05/01/88

Field	Field Name	Type	Width	Dec
1	NOME	Character	40	
2	ENDereco	Character	40	
3	CIDADE	Character	15	
4	ESTADO	Character	2	
5	CEP	Character	6	
6	TEL	Character	14	
7	REM	Memo	10	
** Total **			128	

Figura 11.2.

A seguir, com APPEND, digite os registros sugeridos (os registros são *reais*: Trata-se dos

distribuidores de livros e de *softwares* aplicativos desta Editora em todo o país)

Record #	NOME	ENDERECO
1	DISAL – Distr. Associados de Livros Ltda	R. Vitoria, 486/496
2	Sergio Braga Barbosa	R. Dom Joaquim, 284 – Aldeota
3	Guerreiro & Guerreiro Ltda	R. Treze de Maio, 218 – lj. 2
4	Livro 7 Empreendimentos Culturais Ltda	R. 7 de Setembro, 329
5	LITEC – Livraria Editora Tecnica Ltda	R. dos Timbiras, 257
6	Livraria Importadora Cientifica Ltda	Av. Augusto de Lima, 233 – S/lj. 13/17
7	Editora Itatiaia Ltda	R. Sao Geraldo, 67
8	Editora e Livraria Lunardelli Ltda	Av. Vitor Meirelles, 28
9	CITEC – Ciencia e Tec. Livr. Edit. Ltda	R. Sao Januario, 907 – lj. A
10	Dmerval da Costa Chaves Cia. Ltda	R. Padre Vieira, 9
11	Livraria Grandes Autores Ltda	Av. Antonio Carlos Magalhaes, 260
12	Livraria Tecnica Ltda	CLS – 102 Bloco A – lj. 17
13	Oliveira Costa	R. Joaquim Tavora, 353
14	SAGRA – Liv. Edit. Distribuidora Ltda	R. Joao Alfredo, 448
15	EBRAS – Editora Brasileira Ltda	R. Borges Lagoa, 1044
16	Livraria Cultura Editora Ltda	Av. Paulista, 2073 lj. 153
17	Ciencia Moderna Computacao Ltda	Av. Rio Branco, 156 – Sub-solo 127

CIDADE	ESTADO	CEP	TEL	REM
Sao Paulo	SP	01210	(011)221-1011	Memo
Fortaleza	CE	60110	(085)221-3111	Memo
Curitiba	PR	80510	(041)234-7430	Memo
Recife	PE	51000	(081)231-5213	Memo
Sao Paulo	SP	01208	(011)222-0477	Memo
Belo Horizonte	MG	30190	(031)224-5791	Memo
Belo Horizonte	MG	30150	(031)222-8630	Memo
Florianopolis	SC	88010	(0482)23-0689	Memo
Rio de Janeiro	RJ	20921	(021)580-5480	Memo
Salvador	BA	40020	(071)245-1235	Memo
Salvador	BA	41830	(071)248-2248	Memo
Brasilia	DF	70000	(061)321-4338	Memo
Sao Luiz	MA	65000	(098)222-0107	Memo
Porto Alegre	RS	90050	(0512)27-5222	Memo
Sao Paulo	SP	04038	(011)549-8633	Memo
Sao Paulo	SP	01311	(011)285-4033	Memo
Rio de Janeiro	RJ	20043	(021)262-5723	Memo

(Esta simulação de tela foi dividida em dois devido ao limite físico das páginas deste livro.)

Ao terminar e sair do modo APPEND já teremos nosso novo arquivo LIVRARIA.DBF para interagir com o novo programa (macro) que criaremos a seguir. Digite:

MODIFY COMMAND ETIQUETA ENTER

para entrar numa nova tela de MODIFY COMMAND, agora chamada de ETIQUETA.PRG. Copie o programa exatamente como ele aparece na ilustração do Programa 11.5.

? nome
? endereco

```
? trim(cidade)+'-'+estado
? cep
```

Programa 11.5: Início de ETIQUETA.PRG.

Saia do programa com Ctrl-W, como de hábito e abra seu novo arquivo LIVRARIA.DBF. Acione os comandos:

```
USE LIVRARIA    ENTER
```

e

```
DO ETIQUETA    ENTER
```

Note que a mesma coisa aconteceu no início, quando tentávamos rodar nosso primeiro programa (de duas linhas), ou seja, somente o primeiro registro aparece na tela, o registro de DISAL – Distr. Associados de Livros Ltda., de São Paulo.

Analizando os Procedimentos

Na primeira linha do programa, com “?”, inserimos o campo NOME. Na segunda linha, após o ponto de interrogação, inserimos o campo ENDEREÇO. Logo abaixo inserimos o comando TRIM(CIDADE) que diz ao programa para exibir, na tela ou na impressora, o campo CIDADE *mais* o hífen (“-”), *mais* o campo ao lado (ESTADO). Na última linha programamos o último campo a ser exibido, de acordo com a estética que acabamos de escolher.

Vamos agora voltar à tela de MODIFY COMMAND ETIQUETA.PRG para terminarmos o programa. Digite:

```
MODIFY COMMAND ETIQUETA    ENTER
```

e copie a tela que apresentamos em no Programa 11.6.

```
do while .not. eof( )
? nome
? endereco
? trim(cidade)+'-'+estado
?cep
?
```

Programa 11.6: ETIQUETA.PRG parcial.

Note que este é um programa feito para *processamento direto*. Isto significa que este programa também gera todos os registros de uma só vez, ou seja, que não temos escolha de qual registro poderemos imprimir ou listar na tela. Na seção seguinte veremos como im-

primir essas etiquetas na impressora e depois começaremos a otimizar nosso sistema com vistas a atingir nosso objetivo maior neste capítulo: aprender a programar e construir o já popular “SuperMacro” e o programa controlador `auto@.prg` via arquivos `REF.PRG` e `ETIQUETA.PRG`.

Para imprimir as etiquetas de nosso exemplo, saia da tela de `MODIFY COMMAND` e retorne ao *prompt*. Acione os comandos:

SET TALK OFF ENTER

(para desativar os números dos registros),

USE LIVRARIA ENTER

(para abrir `LIVRARIA.DBF` caso não esteja aberto),

SET PRINT ON ENTER

(para ativar a impressora – que escreverá *tudo* o que for digitado na tela) e

DO ETIQUETA ENTER

(para finalmente visualizar o resultado na tela e na impressora).

Para desativar a impressão via impressora, digite

SET PRINT OFF ENTER

e você poderá digitar qualquer coisa novamente sem o inconveniente de ter de imprimi-la! (Porém, sempre que desejar imprimir etiquetas, você terá de ativar o comando `SET PRINT` novamente.)

Gerando Suas Próprias Etiquetas

Sugerimos que você crie seu próprio arquivo, de acordo com as normas já estabelecidas, para fazer par com seu programa de comandos do tipo `.PRG`. Não é preciso deixar de colocar campos importantes por causa do tamanho do arquivo (como acontece com nosso programa `ETIQUETA`, que não prevê o campo `TEL` que reside no programa fonte `LIVRARIA.DBF` ou o campo `Memo`, que será utilizado mais adiante).

Programa `ETIQUETA.PRG` – Aplicação de Macros

Agora andaremos rapidamente em direção ao “SuperMacro”, otimizando nosso programa `ETIQUETA.PRG`.

Primeiro, entre na tela de ETIQUETA.PRG novamente, com MODIFY COMMAND, e copie todo o programa apresentado no Programa 11.7.

Analizando os Procedimentos

Bem, nosso programa cresceu um pouco, como podemos ver. Novamente, usamos as teclas de controle para acrescentar diversas linhas acima – e algumas abaixo – do programa.

USE LIVRARIA é fundamental para o início do programa. Com este comando, não será mais necessário ter de abrir LIVRARIA.DBF para acessar ETIQUETA.PRG, basta apenas digitar DO ETIQUETA. SET TALK OFF serve para indicar que não queremos imprimir os números dos registros, como já vimos antes; CLEAR indica que o programa deve iniciar sua operação com a tela limpa (isto é, sem se misturar com outras mensagens que porventura se encontrem na tela no momento do processamento); o comando ACCEPT, com a mensagem definida pelo usuário entre aspas (ou apóstrofes), em complemento com a instrução TO *variável*, gera, somente na tela, essa mesma mensagem e prepara a variável “C” para uma comparação na linha seguinte; a próxima linha gera outra mensagem a ser mostrada também somente na tela e define nosso primeiro macro (&C) usado para relacionar a variável C à variável COND ou seja, o macro é substituído pelo conteúdo da variável;

```
***** Programa Gerador de Etiquetas de Enderecamento *****
***** DISTRIBUIDORES *****
```

```
set talk off
clear
accept 'Deseja pesquisar em qual campo?: ' to C
accept 'Procurar qual &C?: ' to COND
accept 'Deseja imprimir? (S/N): ' to R
clear
if upper(R) = 'S'
set print on
endif
go top
do while .not. eof( )
if &C = '&COND'
? nome
? endereco
? trim(cidade)+'-'+estado
? cep
?
?
?
endif
skip
enddo
set print off
return
```

Programa 11.7: ETIQUETA.PRG final.

abaixo mais um comando ACCEPT dá início à linha. Desta vez a mensagem entre aspas relaciona sua respostas a uma nova variável, chamada "R"; na linha seguinte o comando CLEAR novamente limpa a tela para uma execução sem interferências de texto remanescente na tela e precede uma cláusula condicional IF que diz: "se a resposta à pergunta da variável R for S quando transformada em maiúscula (UPPER), então processe os comandos abaixo"; o comando SET PRINT ON abaixo, se atendida a condição acima, liga a impressora; ENDIF termina a condição IF estabelecida anteriormente; GO TOP, como já dissemos antes, faz com que o programa inicie no topo toda vez que ele for rodado; DO WHILE .NOT. EOF(), também já comentado, diz para o programa ser executado enquanto o fim do arquivo não for encontrado; logo abaixo temos outra cláusula IF que diz: "se o macro &C for igual ao macro &COND, então processe o programa abaixo". O restante do programa é o mesmo já visto anteriormente, com o comando SET PRINT OFF no final, para desligar a impressora após a execução (senão tudo o que for digitado será simultaneamente passado para a impressora).

Para testar seu programa, salve tudo com Ctrl-W e, com LIVRARIA.DBF ativo, digite DO ETIQUETA.

Algumas Dicas Importantes

Ainda em tempo: verifique sua digitação. É muito comum, por exemplo, trocar o símbolo "&" pelo cifrão (\$) numa digitação apressada. Também, ao usar as aspas com as mensagens de ACCEPT, para melhor estética do programa, tecle um espaço em branco antes de fechá-lo com " " para evitar que sua resposta fique muito junta ao texto do programa.

Você pode digitar os nomes dos campos com letras maiúsculas ou minúsculas, porém, a digitação dos registros tem que ser idêntica à estética original, ou seja, igual ao que está no programa fonte (por enquanto!).

Analisando Nossos Dois Programas

Até agora, neste capítulo, construímos dois programas .PRG que são programas feitos com MODIFY COMMAND *nome do arquivo*. O primeiro deles, REF.PRG, foi feito para interagir com nosso antigo arquivo, CLIENTES.DBF, que já não tem o mesmo tamanho e os campos que tinha quando o criamos no início do livro. Sugerimos que você preencha o campo TEL, ou seja, coloque os números telefônicos hipotéticos (ou crie seu próprio arquivo CLIENTES com seus próprios dados) em todos os registros que não tenham o número telefônico. Nosso objetivo é melhorar sua capacidade de utilização para nossas futuras interações com o "SuperMacro" que veremos, finalmente, daqui a pouco.

O segundo, programa que criamos neste capítulo foi o arquivo LIVRARIA.DBF, com o objetivo de transformá-lo em banco de dados *fonte* para nosso futuro arquivo de etiquetas de endereçamento ETIQUETAS.PRG. Esse arquivo contém uma lista dos distribuidores

nacionais de livros e/ou *software* aplicativo da EBRAS – Editora Brasileira, com endereços e telefones *reais* (mais um serviço que prestamos a nossos usuários).

Finalmente, criamos nosso terceiro programa, ETIQUETA.PRG, via MODIFY COMMAND, para interagir com LIVRARIA.DBF e imprimir etiquetas de endereçamento, *de uma só vez*. E é aí que está a chave da questão. *De uma só vez!* (Veremos a solução com o programa de controle automático *auto@.prg* e nosso “complexo” de programas “Super-Macro”, mais adiante.)

O usuário já deve saber, obviamente, que pode criar seus próprios programas, com suas próprias informações. Não é preciso gastar tanto papel de impressora – e tempo –, exercitando com nomes e telefones sugeridos pelo autor (e pelo editor dessa edição nacional). Mesmo assim, como divulgadores que somos desse magnífico programa, não temos escolha senão tentar uma orientação modesta – porém sugestiva – e eficaz. Caberá a você, usuário, com os conhecimentos adquiridos e seus próprios meios, melhorar sua capacidade geral de domínio do dBASE III Plus (*ver* Projeto e Desenvolvimento de Programas no Capítulo 13, para programas mais complexos).

auto@ o “SuperMacro”: Controle Automático de Decisões

Finalmente chegamos ao nosso grande objetivo: otimizar/auto@matizar nossos dois programas .PRG e criar um programa controlador que permita que trabalhem com cada programa individualmente ou interligados via controle central (auto@) sem perder nenhuma de suas características.

Usando @, SAY e GET

Os comandos @, SAY e GET são comandos semelhantes aos comandos ?, ACCEPT e TO já usados antes. A diferença básica entre esses comandos é sua complementação com o comando @. @ diz ao programa para exibir as perguntas (os dados) no local indicado pelos números à direita, separados por uma vírgula, que representam localizações “geográficas” da tela, em termos de linhas (as “latitudes”) e colunas (as “longitudes”). Por exemplo, para gerarmos uma mensagem na tela teríamos de digitar: “@ 10,10 SAY ‘Escolha um número. Etiquetas = 1 Tels. = 2 ‘GET NUM’”. A mensagem que digitamos entre aspas *não* resolveria nada! Por quê? Porque primeiro é preciso definir a variável NUM que não está definida no programa! Vamos defini-la agora. Digite:

```
NUM = SPACE(4)    ENTER
```

A seguir digite:

```
CLEAR    ENTER
```

e o comando

```
@ 10,10 say 'Escolha um numero. Etiquetas = 1 Tels. = 2 ' GET NUM ENTER
```

e veja o resultado na tela.

O que você vê é sua mensagem com um espaço destacado (iluminado) para sua resposta (1 ou 2). Mas como responder se o cursor se encontra no *prompt*? A resposta é simples. Está faltando um comando!

Digite, ainda no *prompt*, o comando

```
READ ENTER
```

e você verá que imediatamente seu cursor preencherá o espaço destacado para sua resposta.

Agora digite:

```
DISPLAY MEMORY ENTER
```

para visualizar o conteúdo de sua memória (na figura seguinte exibimos o conteúdo da memória do programa do autor no momento de seu trabalho nesta fase do programa, depois de dezenas de sessões):

```
NUM      puc  C  "      "
      1 variable defined,      6 bytes used
      255 variables available,  5994 bytes available
```

O único problema com esse tipo de nomeação de variável é que as variáveis desaparecem após o desligamento da máquina (mesmo as variáveis *públicas*, como nossa variável NUM). Por isso, inseriremos o comando NUM = SPACE(4) logo no início do programa auto@ para que ele seja lido assim que executarmos o comando DO (DO auto@).

Programando auto@

Vamos iniciar nosso programa auto@. Copie o programa abaixo exatamente como você o vê na ilustração do Programa 11.8. Os comentários serão feitos na seção “Analisando os Procedimentos” a seguir.

Aí está. Um programa simples mas muito eficaz. Principalmente no que se refere à sua flexibilidade, pois com ele poderemos não só controlar nossos dois programas – que ainda serão otimizados –, como também acrescentar mais programas para formar uma rede fantástica de macros e programas utilitários (procure nosso próximo lançamento, dBASE III Plus/dBASE IV – Técnicas Avançadas, que traz o “SuperMacro” autoNet@.prg).


```
*****auto@*** Programa de Controle Automatico (SuperMacro)***
```

```
num = space(4)
clear
@ 10,10 say 'Escolha um numero. Etiquetas = 1 Tels. = 2' get
num
read
if num = '2'
  use clientes
  do ref
endif
if num = '1'
@ 15,1 wait 'Pressione qualquer tecla' to tecla
clear
use livreria
do etiqueta
endif
close databases
```

Programa 11.8: auto@.prg - Programa de Controle automático.

Analisando os Procedimentos

O programa auto@.prg inicia com um título (você pode inserir seu próprio título, se preferir), a definição da variável NUM identificando-a como sendo uma variável *igual* a um espaço alocado para exibição de 4 caracteres (SPACE(4)). Poderíamos, por exemplo, indicar um espaço muito maior, com mais de 200 caracteres em branco e o programa nos reservaria esse espaço para escrevermos qualquer coisa; na segunda linha do programa, colocamos o comando CLEAR, que limpa a tela, como nós bem sabemos; a terceira linha contém nosso comando @ 10,10 SAY visto há pouco. Aí programamos um pequeno menu para o usuário poder optar entre os números 1 e 2, para escolher se prefere acessar o programa ETIQUETA.PRG para gerar etiquetas de endereçamento, ou o programa REF.PRG para acessar sua agenda de nomes e telefones em CLIENTES.DBF. Tudo isso considerando-se que a pergunta deve aparecer no local indicado, ou seja, na posição 10,10; o comando READ faz com que o cursor vá para dentro do espaço iluminado para que possamos escrever o número desejado (experimente retirar READ. Salve tudo com Ctrl-W, rode o programa e veja como o cursor já não vai para essa posição. Volte para o programa com MODIFY COMMAND auto@); a seguir temos a cláusula condicional IF mais uma vez. Agora IF indica que “se a variável NUM for igual a 2, então abra o arquivo CLIENTES, execute o programa REF, termine a condição IF acima”, tudo isso especificado nas três linhas abaixo; a segunda cláusula IF indica que “se o número escolhido for 1, escreva na linha 15, coluna 1, a mensagem definida no comando WAIT (de *esperar*, em inglês) abaixo e aguarde até que pressionemos qualquer tecla (o comando WAIT com a atribuição de uma variável – como TECLA, por exemplo –, não chamada posteriormente, aguarda pelo pressionamento de qualquer tecla (exceto das teclas que não produzem “movimento” de caracteres, como ESC, CTRL, CAPS LOCK, NUM LOCK etc.), não importando o nome de sua variável); abaixo vemos o comando CLEAR novamente, para iniciar o processamento

com uma tela limpa de informações, assim que uma tecla for pressionada no comando anterior. Após o pressionamento de uma tecla qualquer (dentre as já definidas), CLEAR limpa a tela e, imediatamente, abre o arquivo LIVRARIA.DBF e executa o programa ETIQUETA.PRG, terminando também esta condição IF e, conseqüentemente, o programa (que será usado sem terminar, porém, caso os outros dois programas o solicitem – veremos como adiante). Saia da tela de auto@ com Ctrl-W e retorne ao *prompt*.

Conectando REF e auto@

Agora vamos terminar nosso programa REF.PRG, embutindo nele a capacidade de formar uma pequena rede com auto@ e, ao mesmo tempo, deixá-lo trabalhar independentemente. Acione o comando MODIFY COMMAND REF. A seguir, copie as instruções que faltam ao seu programa, exatamente iguais as que você vê no livro, no Programa 11.9.

***** Programa REF - Agenda Eletronica *****

```
use clientes
set talk off
clear
accept 'procurar em que campo?' to C
accept 'procurar qual &C?' to cond
accept 'deseja imprimir? (S/N)' SN
if upper(SN)="S"
    set print on
endif
got top
do while .not. eof( )
    if &C = '&cond'
        ? trim(nome),snome
        ? tel
        ?
        ?
    endif
    skip
enddo
set print off
accept 'Deseja continuar? (S/N):' to R
if upper(R) = 'S'
do auto@
else
clear
endif
```

Programa 11.9: Ref.PRG em rede com auto@.

Analisando os Procedimentos

Repare que REF.PRG tem uma estrutura bem parecida com a estrutura de ETIQUE-

TA.PR.G. Abaixo do título, iniciamos o programa com o comando USE CLIENTES, que nos permite ignorar a abertura do arquivo fonte CLIENTES.DBF, toda vez que iniciarmos nossas operações com REF.PR.G, ou seja, bastará executar REF com o comando DO; o restante das instruções tem o mesmo formato do programa ETIQUETA, com algumas pequenas modificações irrelevantes; após o comando SET PRINT OFF, uma rotina iniciando com o comando ACCEPT, pergunta se o usuário deseja continuar a operação após o término de uma sessão. Se a resposta for positiva (isto é, se for “S” ou “s”, definida com o comando UPPER abaixo), o programa ingressa no programa de controle auto@ que permitirá sua reentrada em REF ou, caso sua opção seja pelo número 1 – dentro de auto@ –, seu acesso a ETIQUETA; caso a resposta seja negativa, o programa limpa a tela e termina a cláusula condicional IF.

Para testar REF e auto@, primeiro digite Ctrl-W para salvar suas modificações e a seguir execute (DO) REF ou auto@ (mas não entre em ETIQUETA, a opção 1 de auto@, ainda).

Após sua interação com auto@ e com REF, ainda no *prompt*, chame sua tela de ETIQUETA.PR.G, com MODIFY COMMAND ETIQUETA. Vamos dar início a nossas alterações finais.

Copie o restante das instruções que faltam para você completar seu programa (cremos não ser necessário comentá-las) do exemplo da ilustração do Programa 11.10.

Para executar o “SuperMacro” auto@, simplesmente salve seu programa com Ctrl-W e digite DO auto@.

O “SuperMacro” auto@ é um exemplo simples, porém didático, do que o usuário pode fazer com a capacidade de programação do dBASE III Plus. Este pequeno complexo de programas pode se tornar uma poderosa ferramenta de trabalho orientado por menus, beneficiando, via programa de controle central, o usuário, que passa a contar com um processo infinitamente mais rápido de operacionalização, como o exemplo do “SuperMacro” autoNet@ criado para nossa edição do dBASE III Plus/dBASE IV – Técnicas Avançadas, que englobará (neste momento estes livros ainda estão em fase de autoria e testes por este autor) uma rede de utilitários *reais* para pronta utilização, além de explicá-los detalhadamente. (*Ver também* Capítulo 13, onde demonstraremos programas mais avançados!)

EDITORAÇÃO DE ARQUIVOS COM O dBASE III Plus: Capacidade Relativa

Vimos anteriormente que nosso arquivo fonte LIVRARIA.DBF possui um campo ainda não explorado, chamado de campo *memo*. Este campo, juntamente com as capacidades de editoração do dBASE III Plus, será tratado objetivamente agora.

O dBASE tem uma capacidade *relativa* de editoração, ou edição, que pode consistir em dois modos de acesso básicos: via MODIFY COMMAND e via campo memo.

Por MODIFY COMMAND, podemos simular uma tela de edição, de maneira semelhante à

```
***** Programa Gerador de Etiquetas de Enderacamento *****
***** DISTRIBUIDORES *****
```

```
use livraria
set talk off
clear
accept 'Deseja pesquisar em qual campo?: ' to C
accept 'Procurar qual &C?: ' to COND
accept 'Deseja imprimir? (S/N): ' to R
clear
if upper (R) = 'S'
set print on
endif
go top
do while .not. eof( )
if &C = '&COND'
? nome
? endereco
trim(cidade) + '-' + estado
? cep
?
?
?
endif
skip
enddo
set print off
accept 'Deseja continuar? (S/N): ' to R
if upper (R) = 'S'
do auto@
else
endif
```

Programa 11.10: Programa Etiquetas.PRG em rede com auto@.

de um processador de texto, porém com uma capacidade restrita a “texto direto”, isto é, pode-se escrever, apagar, copiar, gravar e reformatar texto, mas não há diagramação específica, controle de margem, acentuação etc. Vejamos um breve teste. Acione o comando MODIFY COMMAND DALTON (ou qualquer outro nome de arquivo *novo*, como TEXTO, por exemplo). Em seguida, com a tela vazia de DALTON.PRG, comece a escrever qualquer coisa (o autor deste livro por diversas vezes optou por anotações desse tipo, durante a fase de testes. Sua resumida eficácia se transformou em ganho de velocidade, pois não foi necessária a saída do programa para o processador de texto com tanta frequência).

Já estudamos a tela de MODIFY COMMAND anteriormente neste capítulo, portanto, vamos à próxima etapa. Copie o texto sugerido, se preferir, da Figura 11.3.

Este é um arquivo de texto-teste, para ver as capacidades de “quebra -galhos” do dBASE III Plus. Podemos desde já notar que não temos a capacidade de acentuação, porém, tal qual um bom editor, não precisamos pressionar ENTER a cada fim de linha. Experimente! (Se quiser, você pode teclar ENTER.) Digite Ctrl-W agora, para armazenar este teste.

Faça como o texto mandou, digite Ctrl-W para salvar este arquivo. A seguir, vamos entrar novamente em nosso arquivo para trabalharmos um pouco mais. Digite:

MODIFY COMMAND DALTON ENTER

Agora vamos corrigir alguns erros. Vá com a seta até a palavra “quebra-galhos”. Note que ela não está escrita corretamente. Coloque o cursor no espaço em branco entre a primeira palavra e o hífen e tecele Del, para juntar o hífen à palavra. Em seguida, vá até a palavra “preciamos” e corrija-a para “precisamos”. Qualquer problema quanto a comprimento de frases etc., use o comando Ctrl-KG, para reformatar o parágrafo. Se quiser, salve tudo com Ctrl-W.

Aí está, uma capacidade de editoração resumida mas muito eficaz! .

Bem, vamos tentar progredir um pouco com a editoração – ou *edição*, se você preferir.

Comentaremos agora o campo memo do arquivo LIVRARIA.DBF que criamos especialmente para interagir com nosso “SuperMacro”. Saia da tela de MODIFY COMMAND com Ctrl-W e retorne ao *prompt*. A seguir, digite:

USE LIVRARIA ENTER

(para abrir seu arquivo antes).

Digite o comando

APPEND ENTER

(para entrar no modo de edição)

Use as teclas de controle e vá para o campo memo do registro da EBRAS. Com o cursor sobre o campo memo (REM), pressione Ctrl-PgDn para acessar diretamente a tela de edição do tipo *memo*. Repare que esta tela é idêntica à tela de MODIFY COMMAND, inclusive no que se refere a seus recursos.

A Capacidade do Campo Memo

Um campo do tipo “memo” pode armazenar um total de 5000 caracteres (ao contrário do que diz sua estrutura automática).

Isto quer dizer que podemos escrever à vontade em cada um de nossos campos, e ainda editar com LIST (veremos como daqui a pouco) sua saída. Por exemplo, suponha que você queira escrever a seguinte frase: “Edição de livros de informática de alta qualidade” (ou qualquer outra coisa) no registro da EBRAS. Tudo o que você tem a fazer é digitar a frase

e depois salvá-la em seu novo programa (agora um programa do tipo .DBT, chamado de LIVRARIA.DBT).

Para ver seu campo memo preenchido, é necessário definir os nomes dos campos, isto é, digitar os campos que você quer visualizar *mais* o campo REM – que é o campo memo.

(A vantagem desse sistema para o sistema discutido anteriormente com MODIFY COMMAND, é que com o tipo “memo” seu texto fica vinculado diretamente no arquivo, a cada registro.)

Acione o comando:

LIST NOME, ENDERECO, TEL, REM ENTER

para ver na sua tela os campos pedidos. Como tudo aparece truncado, devido ao seu tamanho, vamos providenciar uma “limpeza” na própria tela. Acione o comando:

LIST OFF NOME, TEL, REM

para que o programa somente considere esses campos ao gerar a tela de LIVRARIA.DBF *sem* (OFF) os números dos registros. Compare o resultado com a Figura 11.4.

Editora Itatiaia Ltda	(031)222-8630	
Editora e Livraria Lunardelli Ltda	(0482)23-0689	
CITEC - Ciencia e Tec. Livr. Edit. Ltda	(021)580-5480	
Dmerval da Costa Chaves Cia. Ltda	(071)245-1235	
Livraria Grandes Autores Ltda	(071)248-2248	
Livraria Tecnica Ltda	(061)321-4338	
Oliveira Costa	(098)222-0107	
SAGRA - Liv. Edit. Distribuidora Ltda	(0512)27-5222	
EBRAS - Editora Brasileira Ltda	(011)549-8633	Edicao de Livros de inf
ormatica de alta qualidade		
Livraria Cultura Editora Ltda	(011)285-4033	
Ciencia Moderna Computacao Ltda	(021)262-5723	

Figura 11.4: Livraria.DBF: acesso ao campo memo por *default*.

Para melhorar ainda mais nossa próxima tela de LIVRARIA com o campo memo, vamos editar seu conteúdo. Acione o comando:

SET MEMOWIDTH TO 20 ENTER

para reduzir o tamanho de nossa mensagem no registro da EBRAS. A seguir, liste o arquivo novamente e veja seu novo resultado na Figura 11.5.

Você pode ajustar o tamanho do campo memo em mais de 10 milhões de caracteres, quanto ao seu tamanho (somente 5000 caracteres serão considerados, de qualquer maneira). O mais importante é o mínimo: 8.

nome	tel	rem
DISAL - Distr. Associados de Livros Ltda	(011)221-1011	
Sergio Braga Barbosa	(085)221-3111	
Guerreiro & Guerreiro Ltda	(041)234-7430	
Livro 7 Empreendimentos Culturais Ltda	(081)231-5213	
LITEC - Livraria Editora Tecnica Ltda	(011)222-0477	
Livraria Importadora Cientifica Ltda	(031)224-5791	
Editora Itatiaia Ltda	(031)222-8630	
Editora e Livraria Lunardelli Ltda	(0482)23-0689	
CITEC - Ciencia e Tec. Livr. Edit. Ltda	(021)580-5480	
Dmerval da Costa Chaves Cia. Ltda	(071)245-1235	
Livraria Grandes Autores Ltda	(071)248-2248	
Livraria Tecnica Ltda	(061)321-4338	
Oliveira Costa	(098)222-0107	
SAGRA - Liv. Edit. Distribuidora Ltda	(0512)27-5222	
EBRAS - Editora Brasileira Ltda	(011)549-8633	Edicao de livros de informatica de alta qualidade
Livraria Cultura Editora Ltda	(011)285-4033	
Ciencia Moderna Computação Ltda	(021)262-5723	

Figura 11.5: Livraria.DBF: acesso ao campo memo programado.

No próximo capítulo trataremos da impressão dos mais variados tipos de relatório. No Capítulo 13 atacaremos Projeto e Desenvolvimento de Programas – incluindo Técnicas de Depuração – e no Capítulo 14 investiremos nosso tempo na criação de telas personalizadas.

Capítulo

12

Impressão de Relatórios Simples e Múltiplos

Neste capítulo concentraremos nossos esforços na impressão dos mais diversos tipos de relatório (cópias impressas) retirados diretamente de arquivos do dBASE III Plus.

Podemos acessar o modo de impressão de duas formas bastante conhecidas por nós: via menu ou diretamente do *prompt*.

Como já estamos há algum tempo trabalhando no *prompt*, iniciaremos nossos comandos dessa forma (entretanto, no dBASE III Plus, o menu que recebemos para trabalhar com a preparação de relatórios é o mesmo para os dois modos – somente o comando *de entrada* difere).

Começaremos por escolher um arquivo já existente, para facilitar as coisas. Primeiro, caso você esteja iniciando suas sessões agora, não se esqueça de atribuir ao acionador B, o acionador que supomos você tenha seu disquete de trabalho (se não estiver operando com um *Winchester*), o *default* de acesso ao acionador. Faça isso com o comando:

SET DEFAULT TO B ENTER

A seguir, abra seu arquivo VENDAS.DBF, o arquivo com o qual trabalharemos neste capítulo. Acione o comando:

USE VENDAS ENTER

A forma mais rápida de impressão de um relatório seria usar a combinação de teclas Shift-PrSc, que, no entanto, geraria qualquer coisa que estivesse na tela, na impressora. O comando *LIST nome do arquivo TO PRINT*, também é uma solução razoável, mas um tanto quanto incompleta. Com este comando, sua exibição de um comando *LIST* apareceria exatamente como na tela, ou seja, sem que você possa interferir na sua estética.

A forma mais eficiente de impressão seria o uso do *Report Generator* do dBASE III Plus, na realidade uma série de comandos com as mais variadas opções para escolha de estética e outros itens de fundamental importância para um relatório.

Com o *default* no acionador B e o arquivo VENDAS.DBF aberto, acione o comando:

CREATE REPORT VENDAS ENTER

ou, se preferir usar o menu

```
MENU
Set Up
Database file
B:
VENDAS.DBF
N
```

para abrir o arquivo e a seleção

MENU

Create

Report

B:

VENDAS

para criar seu relatório.

Bem, neste momento você já deverá ter acessado a tela de opções do *Report Generator*, (Figura 12.1) via *prompt* ou via menu (lembre-se, a tela é a mesma).

Options Groups Columns Locate Exit 01:53:16 am

Page title	TITULO
Page width (positions)	80 ~
Left margin	8
Right margin	0
Lines per page	58
Double space report	No
Page eject before printing	Yes
Page eject after printing	No
Plain page	No

CURSOR <-->	Delete Char: Del	Insert column: ^N	Insert: Ins
Char: <-->	Delete word: ^T	Report Format: F1	Zoom in: ^PgDn
Word: Home End	Delete column: ^U	Abandon: Esc	Zoom out: ^PgUp

MODIFY REPORT	<B:>	B:VENDAS.FRM	Opt: 7/9		
---------------	------	--------------	----------	--	--

Position selection bar -↑↓. Change -↵. Leave menu -←→.
Choose whether or not the printer will form-feed before the report is printed.

Figura 12.1: Menu do Report Generator.

Repare que seu novo arquivo VENDAS é um arquivo com a extensão específica .FRM, portanto ele não se confundirá com um relatório de mesmo nome que tenha uma extensão diferente, como é o caso de VENDAS.DBF.

Ao acessarmos a tela de opções do comando CREATE REPORT, na realidade um menu repleto de alternativas de impressão, vemos suas 5 opções principais: *Options*, *Groups*, *Columns*, *Locate* e *Exit*. Discutiremos todas elas à medida que precisarmos, durante nossas interações neste capítulo. Também, usaremos o formato de apresentação que temos usado quando nos referimos às operações com o menu, considerando que você já deve es-

tar familiarizado com este formato.

Nosso primeiro objetivo será imprimir o arquivo VENDAS.DBF, que, depois de impresso receberá o nome VENDAS.FRM, neste caso. Agora, selecione (antes, porém, leia a seção Analisando os Procedimentos, para entender melhor a seqüência de comandos apresentada):

```
MENU
Options
Page title
TÍTULO
CTRL-END
(ENTER)
(ENTER)
(ENTER)
(ENTER)
```

Analisando os Procedimentos

A primeira seleção do menu *Options, Page title*, nos proporciona uma pequena tela para que possamos escrever o título de nosso relatório. *TÍTULO* significa que podemos escrever o título que desejarmos, desde de que ele caiba no espaço reservado pelo comando *Page title*. *CTRL-END* indica que podemos simplesmente digitar essa combinação de teclas para retornar ao menu *ou* pressionar ENTER tantas vezes quanto forem necessárias (o máximo de vezes chega a quatro, com pelo menos um título escrito), que escolhemos aqui, arbitrariamente, representar com parênteses. Ou seja, toda vez que mostrarmos ENTER no menu escrito entre parênteses, isto indicará que você poderá usar o comando que o precede ou a tecla ENTER, o que lhe convier.

Bem, a seguir, vamos selecionar mais alguns comandos no menu do *Report Generator* para otimizar nosso relatório. Apenas para não desperdiçar papel, selecione a opção *Page eject before printing* com ENTER, até que ela indique “No” (não é necessário digitar “Yes” ou “No”, nesses casos, pois somente o pressionamento de ENTER sobre esses comandos de confirmação basta para alterá-los).

Muito bem. Agora temos de definir *o que* queremos imprimir, *como* imprimir etc.. No momento não queremos usar a opção *Groups* (apesar de estarem uma ao lado da outra, as opções não têm um relacionamento seqüencial necessário). Passemos então para a opção *Columns*, para iniciarmos a seleção dos títulos de cada coluna de nosso antigo arquivo VENDAS.DBF, que, ao iniciarmos esta operação de impressão, recebeu o nome de VENDAS.FRM (não é necessário dar o nome do arquivo-relatório idêntico ao nome do arquivo-fonte deste mesmo relatório. Por exemplo, poderíamos ter chamado este relatório de RELAT ou qualquer outro *nome de arquivo* e ele ainda assim seria feito com base em seu arquivo-fonte, VENDAS.DBF, aberto no início de nossas operações com o *Report Generator*). Na opção principal *Columns*, selecione:

```

MENU
Columns
Contents
ENTER
F10
CLIENTE
ENTER
Heading
ENTER
Cliente
CTRL-END
PG DN

```

Analisando os Procedimentos

Primeiro selecionamos a opção principal *Columns*, para iniciarmos nossa edição das colunas do relatório a ser impresso. Neste menu, selecionamos a subopção *Contents* que irá definir qual conteúdo de cada coluna iremos acessar. Por exemplo, para imprimirmos nosso relatório do arquivo VENDAS.DBF, no modo do menu *Report Generator*, após as seleções prévias feitas na opção *Options* já comentadas, entramos na subopção *Contents* da opção principal *Columns* para escolher os campos do arquivo VENDAS.DBF que serão impressos, bem como os títulos de cada campo, que poderão ou não serem os mesmos, desde que sejam digitados via subopção *Heading* (cabeçalho). Aqui, colocamos o cabeçalho da coluna CLIENTE em letras minúsculas para que você possa comparar seus resultados com mais facilidade após nossa primeira impressão. Não se esqueça de ir para a subopção *Heading* com a seta para baixo (no exemplo do menu acima, já colocamos o nome da opção – e não a seta). Em seguida digitamos a combinação Ctrl-End para sair da tela especial de *Heading* (ou ENTER 4 vezes) e pressionamos PgDn para trocar de “página” e irmos diretamente para a seleção de outro campo, nesse mesmo menu – e assim sucessivamente –, até que estejamos satisfeitos com o conteúdo do relatório.

É importante notar também que, abaixo dessas duas opções, *Contents* e *Heading*, vemos outras três opções – ou subopções, se preferir, que são: *Width*, *Decimal Places* e *Total this column*. Repare que a opção *Width* é incrementada de um número cada vez que escolhemos um campo. O dBASE faz isso automaticamente para alocar seus espaços no relatório. Veremos as outras duas opções mais adiante. Agora, vamos continuar escolhendo os campos que serão impressos (optamos por todos os campos do arquivo VENDAS.DBF, para garantir que você seja treinado adequadamente!).

Selecione, ainda no menu *Columns*, em sua nova página:

```

MENU
Contents
ENTER
F10

```

PRODUTO
ENTER
Heading
ENTER
Produto
CTRL-END
PG DN

Analisando os Procedimentos

Nossos procedimentos aqui foram idênticos aos anteriores, sendo que utilizamos o nome do campo **PRODUTO**, que vem a ser o segundo campo de nosso relatório (podemos indicar o campo que desejarmos). Portanto, para maiores esclarecimentos, refira-se aos comentários de *Analisando os Procedimentos* anterior.

Repita a mesma operação para os outros campos que faltam, **QTDE**, **OTN**, **VENC** e **STATUS**. Ao terminar de preencher todos os campos, sempre escrevendo na pequena tela especial de *Heading* o mesmo nome do campo fonte em minúsculas, use a seta à direita para ir direto para a opção **Exit** e pressione **ENTER** sobre a subopção **Save**, para salvar seu relatório chamado **VENDAS.FRM** – o relatório editado de nosso arquivo **VENDAS.DBF**!

A seguir, vamos à impressão propriamente dita e aos comentários subsequentes.

Acione o comando (supondo que você esteja no *prompt*):

REPORT FORM VENDAS TO PRINT ENTER

ou, caso prefira o menu:

MENU
Retrieve
Report
B:
VENDAS
Y

e você verá seu relatório na tela e na impressora (se ela estiver ligada adequadamente), como o exemplo da Figura 12.2.

Repare que nosso **TÍTULO** do relatório não está centralizado e que os títulos das colunas (campos) encontram-se em letras minúsculas exceto pelas letras iniciais, de acordo com o que programamos. Mas podemos melhorar muito nosso relatório, inclusive acrescentando detalhes que o transformarão em um relatório mais útil. Por exemplo, note que nosso relatório foi exibido com as mesmas características de uma impressão com **LIST TO PRINT** do próprio arquivo **VENDAS.DBF**, com pequenas modificações, como os títulos das colu-

Page No. 1
06/28/88

TITULO					
Ciente	Produto	Qtde	Otn	Venc	Status
001	LOTUS T.A.	40	4.00	07/20/88	.T.
002	WordLegal	60	15.00	07/20/88	.T.
003	WordLegal	20	15.00	07/20/88	.T.
003	dBASE III	15	6.00	07/20/88	.T.
001	APPLE	10	4.00	07/18/88	.T.
010	ASMIBM PC	80	6.00	07/18/88	.T.
042	ASMIBM PC	25	6.00	07/17/88	.T.
009	LOTUS T.A.	8	4.00	07/12/88	.T.
033	WordStar	80	4.00	07/08/88	.T.
012	MKT	80	4.00	06/30/88	.F.
005	MKT	10	4.00	06/30/88	.F.
008	MATFIN	80	4.00	06/22/88	.F.
014	WordLegal	40	15.00	06/22/88	.F.
021	dBASE III	20	6.00	06/18/88	.F.
022	dBASE III	10	6.00	06/18/88	.F.
038	WordLegal	30	6.00	06/12/88	.F.
037	ASMIBM PC	5	6.00	06/08/88	.F.
001	dBASE III	80	6.00	06/08/88	.F.
007	WordLegal	20	4.00	06/06/88	.F.
019	WordLegal	60	6.00	06/04/88	.F.
*** Total ***		773	131.00		

Figura 12.2: VENDAS.FRM: relatório personalizado de VENDAS.DBF.

nas dos campos, a data da impressão (neste caso a data *atual* será a data que determinarmos durante o processo de inicialização) e o total dos campos Qtde e Otn. Como você já deve ter percebido, todo esse trabalho para gerar um relatório quase igual ao próprio arquivo fonte não serve para muita coisa. Pois é isso que vamos tentar resolver agora, nesta próxima segunda sessão com o *Report Generator*.

Para retornar ao mesmo arquivo de impressão, VENDAS.FRM, acione o comando:

MODIFY REPORT VENDAS ENTER

ou, via menu:

MENU
M*odify*
*R***eport**
B:
VENDAS

e você terá de volta seu arquivo para, na coluna *Options*, reeditá-lo.

Preencha a opção *Page title* com um novo título, chamado TÍTULO DO RELATÓRIO.

Para que este título fique centralizado, passe para a opção *Page Width (positions)* e pressione ENTER para depois inserir um novo número. Coloque 60, para diminuir a largura lógica do papel de 80 para 60 colunas, de modo que o título saia centralizado (senão seu título ficaria centralizado no meio de uma página de 80 colunas, ou seja mais à direita do nosso relatório). Pressione ENTER para confirmar a alteração. Note também que nossa primeira impressão ficou muito próxima à margem esquerda do papel. Vamos melhorar isso, passando para a opção *Left margin*, pressionando ENTER e inserindo 10, no lugar de 8, que é o *default* da margem esquerda (pode-se também usar as setas para cima e para baixo para aumentar ou diminuir o valor de um número). Como nosso texto é pequeno, não precisaremos mexer na margem direita. Também não será necessário mexer na quantidade de linhas por página, nossa próxima opção. Em seguida, como não queremos um relatório com espaços duplos neste exemplo, passemos então para a opção seguinte, *Page eject before printing*. Pressione ENTER para escolher *Yes*, pois queremos uma folha nova para nosso exemplo de impressão editada. Caso você queira um avanço de papel imediatamente após a impressão, insira *Yes* na próxima opção, *Page eject after printing*. Nesse nosso novo relatório, também queremos uma “página pura”, ou seja, sem a data atual, portanto, tecle *Yes* sobre esta opção.

Após as alterações sob a opção *Options*, passe para a opção *Columns* para mais uma pequena alteração.

Options	Groups	Columns	Locate	Exit 02:09:50 am
Page title		TITULO D		
Page width (positions)		60		
Left margin		10		
Right margin		0		
Lines per page		58		
Double space report		No		
Page eject before printing		Yes		
Page eject after printing		Yes		
Plain page		Yes		

Figura 12.3: Primeiras alterações em *Options*.

De acordo com as operações já vistas e praticadas neste capítulo, insira novamente os nomes dos campos na opção *Heading*, desta vez em letras maiúsculas, sobre os nomes existentes. A seguir, saia do modo de impressão com *Exit* e *Save* e gere o relatório na impressora. Compare com os resultados da Figura 12.4.

Há mais um detalhe de fundamental importância. Você pode usar seu arquivo indexado para que a saída impressa seja ainda mais objetiva. Por exemplo, para gerar nosso arquivo VENDAS.FRM indexado pelo campo CLIENTE a partir do *prompt*, simplesmente acione o comando:

INDEX ON CLIENTE TO CLIENTE ENTER

TITULO DO RELATORIO					
CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
001	LOTUS T.A.	40	4.00	07/20/88	.T.
002	WordLegal	60	15.00	07/20/88	.T.
003	WordLegal	20	15.00	07/20/88	.T.
003	dBASE III	15	6.00	07/20/88	.T.
001	APPLE	10	4.00	07/18/88	.T.
010	ASMIBM PC	80	6.00	07/18/88	.T.
042	ASMIBM PC	25	6.00	07/17/88	.T.
009	LOTUS T.A.	8	4.00	07/12/88	.T.
033	WordStar	80	4.00	07/08/88	.T.
012	MKT	80	4.00	06/30/88	.F.
005	MKT	10	4.00	06/30/88	.F.
008	MATFIN	80	4.00	06/22/88	.F.
014	WordLegal	40	15.00	06/22/88	.F.
021	dBASE III	20	6.00	06/18/88	.F.
022	dBASE III	10	6.00	06/18/88	.F.
038	WordLegal	30	6.00	06/12/88	.F.
037	ASMIBM PC	5	6.00	06/08/88	.F.
001	dBASE III	80	6.00	06/08/88	.F.
007	WordLegal	20	4.00	06/06/88	.F.
019	WordLegal	60	6.00	06/04/88	.F.
*** Total ***		773	131.00		

Figura 12.4: VENDAS.FKM reformatado.

e comande sua impressão em seguida, com REPORT FORM VENDAS TO PRINT ou, preferindo o menu, selecione:

```

MENU
Set Up
Database file
B:
VENDAS.DBF
Y
VENDAS.NDX
<-
Retrieve
Report
B:
VENDAS.FRM
Execute the command
Y

```

Ou seja, via menu também podemos executar a mesma tarefa caso já tenhamos um arquivo VENDAS.NDX indexado anteriormente pelo campo CLIENTE (ou por qualquer outro campo, quando for o caso). Compare seu novo resultado com o campo CLIENTE indexado na Figura 12.5.

TITULO DO RELATORIO					
CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
001	LOTUS T.A.	40	4.00	07/20/88	.T.
001	APPLE	10	4.00	07/18/88	.T.
001	dBASE III	80	6.00	06/08/88	.F.
002	WordLegal	60	15.00	07/20/88	.T.
003	WordLegal	20	15.00	07/20/88	.T.
003	dBASE III	15	6.00	07/20/88	.T.
005	MKT	10	4.00	06/30/88	.F.
007	WordLegal	20	4.00	06/06/88	.F.
008	MATFIN	80	4.00	06/22/88	.F.
009	LOTUS T.A.	8	4.00	07/12/88	.T.
010	ASMIBM PC	80	6.00	07/18/88	.T.
012	MKT	80	4.00	06/30/88	.F.
014	WordLegal	40	15.00	06/22/88	.F.
019	WordLegal	60	6.00	06/04/88	.F.
021	dBASE III	20	6.00	06/18/88	.F.
022	dBASE III	10	6.00	06/18/88	.F.
033	WordStar	80	4.00	07/08/88	.T.
037	ASMIBM PC	5	6.00	06/08/88	.F.
038	WordLegal	30	6.00	06/12/88	.F.
042	ASMIBM PC	25	6.00	07/17/88	.T.
*** Total ***		773	131.00		

Figura 12.5: VENDAS.FKM indexado.

As Outras Opções de Report

As outras duas opções com as quais ainda não operamos são **Groups** e **Locate**. Vamos a elas agora.

Entre novamente no menu do *Report Generator*, via **MODIFY** (ou **Modify**, do menu, se preferir). Acesse a coluna **Groups**. Agora, vamos trabalhar com subtotais por grupos selecionados no menu.

Selecione:

```

MENU
Groups
Group on expression
ENTER
F10
PRODUTO
ENTER
Group heading
ENTER

```

Produto
ENTER

Agora entre na opção principal *Options* e altere, como exercício, as opções *Page eject before printing* (porém, reserve uma folha em branco para iniciar sua impressão com uma página nova) para *No* e *Plain page*, também para *No*, para obtermos uma página com marcação numérica e data do sistema operacional.

Desta vez não discutiremos os procedimentos. Faça sua comparação entre os comandos que acabou de inserir e o resultado da impressão. Salve tudo com *Exit* e *Save* antes.

Ainda antes de gerar o relatório, indexe o campo PRODUTO do arquivo VENDAS.DBF com a opção *Index* do menu *Organize* ou a partir do *prompt* com o comando INDEX.

Para gerar o relatório, digite:

REPORT FORM VENDAS TO PRINT

ou, se preferir o menu:

MENU
Retrieve
Report
B:
VENDAS.DBF
Y

Veja o resultado de seu trabalho (aqui apresentamos um resultado parcial por razões de estética. Seu resultado deverá ocupar um pouco mais de uma folha de formulário contínuo impressa – a página 2 –, com o título e os campos se repetindo e os totais gerais dos campos QTDE e OTN.) e compare com o da primeira impressão. Observe que ainda temos o título “TÍTULO DO RELATÓRIO” e os nomes dos campos que definimos anteriormente no menu *Columns*. Você pode retirá-los, ou editá-los, retomando ao menu do *Report Generator*, à opção *Columns* e redigitando as condições desejadas. É importante você saber também, que escolhemos o campo PRODUTO para indexar e preparar nosso relatório porque este é o campo que, quando gerado, apresenta mais condensação, ou seja, é o que mais se agrupa e o que menos consome papel (o tamanho do relatório). Tente, como exemplo, fazer o mesmo relatório pelo campo CLIENTE. Depois compare o tamanho dos dois e veja a diferença.

Nosso relatório indexado pelo campo PRODUTO deverá estar como o exemplo da Figura 12.6 (muito provavelmente ele consumirá um pouco mais de uma página. Experimente colocar a impressora no modo de caracteres comprimidos para melhorar a aparência do relatório).

Page No.. 1
06/28/88

TITULO DO RELATORIO

CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
** Produto APPLE					
001	APPLE	10	4.00	07/18/88	.T.
** Subtotal **		10	4.00		
** Produto ASMIBM PC					
010	ASMIBM PC	80	6.00	07/18/88	.T.
042	ASMIBM PC	25	6.00	07/17/88	.T.
037	ASMIBM PC	5	6.00	06/08/88	.F.
** Subtotal **		110	18.00		
** Produto LOTUS T.A.					
001	LOTUS T.A.	40	4.00	07/20/88	.T.
009	LOTUS T.A.	8	4.00	07/12/88	.T.
** Subtotal **		48	8.00		
** Produto MATFIN					
008	MATFIN	80	4.00	06/22/88	.F.
** Subtotal **		80	4.00		
** Produto MKT					
012	MKT	80	4.00	06/30/88	.F.
005	MKT	10	4.00	06/30/88	.F.
** Subtotal **		90	8.00		
** Produto WordLegal					
002	WordLegal	60	15.00	07/20/88	.T.
003	WordLegal	20	15.00	07/20/88	.T.
014	WordLegal	40	15.00	06/22/88	.F.
038	WordLegal	30	6.00	06/12/88	.F.
007	WordLegal	20	4.00	06/06/88	.F.
019	WordLegal	60	6.00	06/04/88	.F.
** Subtotal **		230	61.00		
** Produto WordStar					
033	WordStar	80	4.00	07/08/88	.T.
** Subtotal **		80	4.00		

Figura 12.6: VENDAS.FRM gerado com a opção *Groups*.

Como prática, tente usar as outras subopções de **Groups**. Elas são fáceis de usar e não necessitam de maiores detalhes quanto a seu tratamento didático. Além disso, esse exercício certamente lhe será benéfico.

Passemos então para a opção **Locate**.

Retorne ao menu do *Report Generator* com **MODIFY REPORT VENDAS**, ou via menu. A seguir, entre na opção **Locate** com a seta ou pressionando a primeira letra do comando (L) e pressione **ENTER**. Selecione o campo **CLIENTE**, o campo que estamos tentando localizar para executar uma alteração. Note que a seleção de **CLIENTE** nos leva imediatamente à opção **Columns**, que nos apresenta nossas próprias definições anteriores do campo **CLIENTE**. É só isso! **Locate** simplesmente nos ajuda a acessar um determinado campo existente no *Report Generator* para edição adicional.

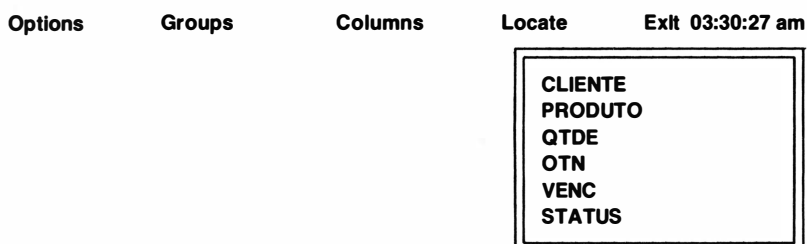


Figura 12.7: Menu Principal de Locate/LIVRARIA.DBF.

Bem, para pelo menos executarmos alguma tarefa, vá para a subopção **Width** e altere seu número para 9 (o máximo no relatório que construímos até agora seria 10). Salve tudo com **Exit** e **Save**. Antes de imprimir o relatório, verifique se o seu arquivo está indexado pelo campo **PRODUTO**, tal qual da vez anterior. Depois gere seu relatório modificado. Compare-o com o relatório anterior e veja que agora os nomes dos produtos encontram-se mais à direita, exatamente abaixo do nome do produto que encabeça a lista (novamente, exibimos, por razões de estética, um resultado parcial, ou seja, do início do relatório até o limite físico da página deste livro). Veja Figura 12.8.

É óbvio que podemos fazer modificações mais complicadas, mas isso não é necessário aqui, pois esperamos que este treinamento lhe proporcione as condições básicas para sua evolução natural.

PRODUÇÃO RÁPIDA DE ETIQUETAS VIA MENU

Lembra-se de nosso “SuperMacro” do capítulo anterior? Pois bem, agora vamos usar um de seus programas, o de geração de etiquetas de endereçamento, que contém os endereços reais dos distribuidores desta Editora em todo o Brasil.

Primeiro, abra o arquivo que contém os dados com os quais trabalharemos, o arquivo fonte **LIVRARIA.DBF**. A seguir, vá para a opção **Create** do menu principal e selecione:

Page No. 1
06/28/88

TITULO DO RELATORIO

CLIENTE	PRODUTO	QTDE	OTN	VENC	STATUS
** Produto APPLE					
001	APPLE	10	4.00	07/18/88	.T.
** Subtotal **		10	4.00		
** Produto ASMIBM PC					
010	ASMIBM PC	80	6.00	07/18/88	.T.
042	ASMIBM PC	25	6.00	07/17/88	.T.
037	ASMIBM PC	5	6.00	06/08/88	.F.
** Subtotal **		110	18.00		
** Produto LOTUS T.A.					
001	LOTUS T.A.	40	4.00	07/20/88	.T.
009	LOTUS T.A.	8	4.00	07/12/88	.T.
** Subtotal **		48	8.00		
** Produto MATFIN					
008	MATFIN	80	4.00	06/22/88	.F.
** Subtotal **		80	4.00		
** Produto MKT					
012	MKT	80	4.00	06/30/88	.F.
005	MKT	10	4.00	06/30/88	.F.
** Subtotal **		90	8.00		
** Produto WordLegal					
002	WordLegal	60	15.00	07/20/88	.T.
003	WordLegal	20	15.00	07/20/88	.T.
014	WordLegal	40	15.00	06/22/88	.F.
038	WordLegal	30	6.00	06/12/88	.F.
007	WordLegal	20	4.00	06/06/88	.F.
019	WordLegal	60	6.00	06/04/88	.F.
** Subtotal **		230	61.00		
** Produto WordStar					
033	WordStar	80	4.00	07/08/88	.T.
** Subtotal **		80	4.00		

Figura 12.8: VENDAS.FRM gerado com a opção *Groups*.

MENU*Create**Label**B:***ENDERECA****ENTER**

Agora, podemos ver o menu principal de *Label*, que consiste em três opções: *Options*, *Contents* e *Exit*. Na primeira opção, *Options*, podemos ver a subopção *Predefined size*:. Esta opção indica os tamanhos pré-determinados do programa, que são quatro: *3 1/2 x 15/16 by 1*, *3 1/2 x 15/16 by 2*, *3 1/2 x 15/16 by 3*, *3 2/10 x 11/12 by 3 (Cheshire)* e, finalmente, *4 x 1 7/16 by 1* – que usaremos em nossa impressão.

Para selecionar esses tamanhos, simplesmente tecle ENTER sobre esta opção até que seu tamanho apareça na tela. Escolha o maior (maior largura), ou seja, *4 x 1 7/16 by 1*.

Copie o restante das informações da tela a seguir. Note que podemos escolher nossos próprios padrões: em *Label width*: podemos escolher o tamanho de nossa etiqueta; em *Label height*: podemos escolher a altura de cada uma; em *Left margin*: podemos finalmente escolher a margem esquerda, ou seja, o início a partir da margem esquerda, da impressão de nossas etiquetas (quando trabalhamos com o “SuperMacro” não tínhamos essa opção); em *Lines between labels*: podemos optar pela quantidade de linhas entre as etiquetas; em *Spaces between labels*: podemos colocar a quantidade de espaço que queremos entre as etiquetas e, finalmente, em *Labels across page*: podemos escolher quantas colunas de etiquetas desejamos imprimir na mesma folha (1 = uma “fila” de etiquetas, uma sobre a outra). Vá para essa opção, pressione ENTER e, com a seta ou com o teclado numérico, insira o número 2, pois nesse exemplo imprimiremos duas colunas de etiquetas de endereçamento (são um máximo de cinco colunas, de 1 a 5).

Sua tela deverá estar como na Figura 12.9.

Options

Contents

Exit 11:31:43 am

Predefined size:	4 x 1 7/16 by 1
Label width:	40
Label height:	8
Left margin:	0
Lines between labels:	1
Spaces between labels:	0
Labels across page:	2

Figura 12.9: Menu de formatação de relatórios.

Agora passe para a opção *Contents* do menu principal. Observe que aqui temos um quadro numerado de 1 a 5, reservado especialmente para nossa programação. Como iremos extrair

os dados de nossas etiquetas de endereçamento do arquivo *LIVRARIA.DBF* – que inclusive já foi aberto no início desta sessão –, vamos copiar o *núcleo* de nosso programa de geração de etiquetas que já preparamos para interações com o “SuperMacro” *auto@*.

Após copiar o programa, sua tela deverá ficar como na Figura 12.10.

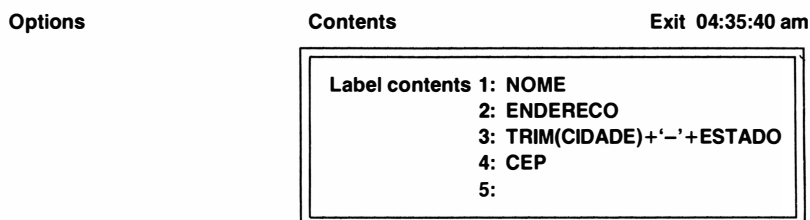


Figura 12.10: Programação de campos por etiquetas de endereçamento.

Um detalhe importante: Você só pode preencher a pequena tela de *Contents* com até 8 linhas de programa, portanto, coloque um programa que tenha começo e fim! (No momento o melhor seria copiar nossa sugestão...).

Note que nossa tela com o programa apresenta o núcleo do programa de geração de etiquetas feito no capítulo anterior, ou seja, o *mínimo* de linhas necessárias para que essas etiquetas sejam impressas adequadamente.

Salve seu trabalho com *Exit* e *Save*. A seguir vá para o menu *Retrieve* e para a opção *Label*. Selecione o acionador onde se encontra seu disquete de trabalho (provavelmente B:) e depois chame seu novo arquivo *ENDEREÇA.LBL* com *Retrieve*, *Label*, *B:*, *ENDEREÇA.LBL*, *Execute the command* e *Y* para gerá-lo na impressora (ou *N* caso prefira uma exibição somente na tela).

Observe o início de sua impressão de etiquetas de endereçamento (na Figura 12.11 novamente mostramos somente as 10 primeiras em seu formato original).

Podemos ver que elas não estão em nenhuma ordem específica. Isto acontecerá se você não indexar seu arquivo pelo campo desejado. Por exemplo, para gerar estas mesmas etiquetas ordenadas pelo campo *NOME*, use o comando *Index* do menu *Organize* (ou diretamente do *prompt*). Se tiver alguma dúvida quanto a sua sintaxe, recorra ao menu para facilitar as coisas. Nesse caso, antes de sair da tela do menu, verifique a linha de comando que se forma automaticamente à medida que selecionamos as opções do comando.

Sua tela com as etiquetas pela ordem do campo *NOME* deverá ficar como no exemplo da Figura 12.12.

Em tempo: sua única saída para melhorar a aparência quase que truncada de alguns nomes de distribuidores (como o registro da *DISAL*, por exemplo), será reformatar esses nomes, abreviando-os, por exemplo, para que eles não fiquem tão próximos um do outro.

DISAL – Distr. Associados de Livros Ltda R. Vitoria, 486/496 Sao Paulo-SP 01210	Sergio Braga Barbosa R. Dom Joaquim, 284 – Aldeota Fortaleza-CE 60110
Guerreiro & Guerreiro Ltda R. Treze de Maio, 218 – lj. 2 Curitiba-PR 80510	Livro 7 Empreendimentos Culturais Ltda R. 7 de Setembro, 329 Recife-PE 51000
LITEC – Livraria Editora Tecnica Ltda R. dos Timbiras, 257 Sao Paulo-SP 01208	Livraria Importadora Cientifica Ltda Av. Augusto de Lima, 233 – S/lj. 13/17 Belo Horizonte-MG 30190
Editora Itatiaia Ltda R. Sao Geraldo, 67 Belo Horizonte-MG 30150	Editora e Livraria Lunardelli Ltda Av. Vitor Meirelles, 28 Florianopolis-SC 88010
CITEC – Ciencia e Tec. Livr. Edit. Ltda R. Sao Januario, 907 – lj. A Rio de Janeiro-RJ 20921	Dmerval da Costa Chaves Cia. Ltda R. Padre Vieira, 9 Salvador-BA 40020

Figura 12.11: Etiquetas impressas de ENDEREÇA.LBL.

CITEC – Ciencia e Tec. Livr. Edit. Ltda R. Sao Januario, 907 – lj. A Rio de Janeiro-RJ 20921	Ciencia Moderna Computacao Ltda Av. Rio Branco, 156 – Sub-solo127 Rio de Janeiro-RJ 20043
DISAL – Distr. Associados de Livros Ltda R. Vitoria, 486/496 Sao Paulo-SP 01210	Dmerval da Costa Chaves Cia. Ltda R. Padre Vieira, 9 Salvador-BA 40020
EBRAS – Editora Brasileira Ltda R. Borges Lagoa, 1044 Sao Paulo-SP 04038	Editora Itatiaia Ltda R. Sao Geraldo, 67 Belo Horizonte-MG 30150
Editora e Livraria Lunardelli Ltda Av. Vitor Meirelles, 28 Florianopolis-SC 88010	Guerreiro & Guerreiro Ltda R. Treze de Maio, 218 – lj. 2 Curitiba-PR 80510
LITEC – Livraria Editora Tecnica Ltda R. dos Timbiras, 257 Sao Paulo-SP 01208	Livraria Cultura Editora Ltda Av. Paulista, 2073 Lj. 153 Sao Paulo-SP 01311

Figura 12.12: Etiquetas de ENDEREÇA.LBL pelo campo Nome.

Impressão Condicional

Podemos também imprimir etiquetas que atendam a determinadas condições especificadas previamente no menu. Por exemplo, para imprimirmos somente as etiquetas dos distribuidores do estado de São Paulo, selecione:

```

MENU
Retrieve
Label
B:
ENDERECA.LBL
Build a search condition
ESTADO
= Equal To
SP
No more conditions
Execute the command
Y

```

e sua tela e formulário contínuo ficarão como na Figura 12.13.

DISAL – Distr. Associados de Livros Ltda	LITEC – Livraria Editora Tecnica Ltda
R. Vitoria, 486/496	R. dos Timbiras, 257
Sao Paulo-SP	Sao Paulo-SP
01210	01208
EBRAS – Editora Brasileira Ltda	Livraria Cultura Editora Ltda
R. Borges Lagoa, 1044	Av. Paulista, 2073 Lj. 153
Sao Paulo-SP	Sao Paulo-SP
04038	01311

Figura 12.13: Etiquetas somente de SP.

As capacidades do menu são as mesmas, portanto use sua criatividade – e sua necessidade – para executar seu trabalho de acordo com suas próprias especificações.

Lembre-se de que também é possível fazer uma impressão rápida via combinação de teclas Shift-PrSc, porém essa impressão é indiscriminada, ou seja, *tudo* o que *estiver* na tela será gerado na impressora.

Impressão Via Prompt

A impressão via *prompt* é muito simples. Por exemplo, suponha que você queira imprimir as etiquetas do último exemplo feito com o menu. Simplesmente você teria de digitar:

```
LABEL FORM ENDERECA FOR ESTADO = 'SP' ENTER
```

(não inclua o nome do acionador se você já estiver no acionador que contém o disco de trabalho, ou o dBASE considerará a letra excedente como um nome de arquivo que, conseqüentemente, não existirá).

Se você simplesmente quiser gerar todo o arquivo de etiquetas, digite:

LABEL FORM ENDERECA TO PRINT ENTER

ou, se quiser uma visualização somente na tela, omita o comando TO PRINT:

LABEL FORM ENDERECA ENTER

Bem, vamos ficando por aqui nesta sessão. Aproveite e mude a margem esquerda de seu arquivo ENDEREÇA.LBL que está imprimindo as etiquetas muito próximas da margem (*Modify*, *Label* etc.). Vamos à próxima sessão de impressão, desta vez usando arquivos múltiplos (porém, caso você não tenha passado pelo Capítulo 10, quando aprendemos a operar com arquivos múltiplos, faça isso agora!).

Impressão de Arquivos Múltiplos

No Capítulo 10 trabalhamos com bancos de dados múltiplos, via arquivos do tipo *View*. Com essa técnica aprendemos a juntar dois arquivos num arquivo só (dividindo o arquivo VENDAS.DBF) e acabamos por criar os arquivos GRANA.DBF, PERFIL.DBF e, finalmente, VNOVO.VUE, o arquivo *View* criado a partir desses dois últimos.

Nesta seção, trataremos de imprimir esse arquivo na impressora, já que no Capítulo 10 nosso objetivo maior foi aprender a gerá-lo na tela.

Para gerar um arquivo múltiplo (conseqüentemente, do tipo *View*) numa impressora, primeiro temos de abrir o arquivo *View*. Passe para a opção *Set Up* do menu *Assist* e selecione:

```
MENU
Set Up
View
B:
VNOVO
```

ou, se preferir usar os comandos digitados diretamente do *prompt*:

SET DEFAULT TO B ENTER

(caso seu acionador *default* ainda não seja o B);

SET VIEW TO VNOVO ENTER

(o arquivo múltiplo que criamos anteriormente, e que será o arquivo trabalhado para ser transformado em relatório).

A seguir, considerando que estamos no menu, passe para a opção **Create** e selecione:

MENU

Create

Report

B:

REVIEW

ENTER

ou, diretamente do *prompt*:

CREATE REPORT REVIEW ENTER

Analizando os Procedimentos

Primeiro abrimos o antigo arquivo VNOVO (de *View* + NOVO) que construímos no Capítulo 10, composto dos arquivos GRANA e PERFIL. A seguir criamos nosso relatório que, por sua vez, também recebe um nome específico que aqui escolhemos arbitrariamente como sendo REVIEW.

Após esses procedimentos, devemos acessar a tela do *Report Generator*, da mesma forma que a acessamos durante boa parte deste capítulo.

A Opção Options

Agora já sabemos mais ou menos o que fazer: no menu *Options*, escolha a opção desejada. Porém, lembre-se de que um tamanho de página de 80 para um tamanho de margem esquerda de 8 é ideal para impressões com a impressora no estado normal de impressão, isto é, sem caracteres comprimidos (e sem centralização do título). Para centralizar o título do relatório e para centralizar uma impressão com caracteres comprimidos é necessário editar essas opções colocando, por exemplo, 100 para o tamanho da página e 60 para o tamanho da margem esquerda. As outras opções você já conhece bem (senão, reporte-se ao início do capítulo para familiarizar-se).

A Opção Groups

Passemos então para a opção seguinte, *Groups*. Aqui veremos algumas pequenas diferenças em relação aos relatórios que fizemos até agora. Pressione ENTER sobre a subopção *Group on expression*. Em seguida, tecla F10 para escolher um campo em comum entre os dois arquivos que serão impressos formando um único arquivo (como fizemos com *View*

no Capítulo 10). Escolha o campo **CLIENTE**, pressionando **ENTER** sobre esta opção seguido de outro **ENTER** para confirmar a entrada na posição de seleção. Passe para a opção abaixo, *Heading*, pressione **ENTER** para selecioná-la, seguido da digitação do nome desejado. Digite *cliente*, assim mesmo, em letras minúsculas, para facilitar sua comparação posterior (ou seja, para que você saiba distinguir o que você digitou do que já estava no programa e apenas foi editado).

A Opção Columns

Na opção *Contents*, novamente pressione **ENTER** para confirmá-la e **F10** para acessar a lista de campos oferecida pelo menu. Escolha o campo **OTN**, pressionando **ENTER** sobre ela seguido de outro **ENTER** para confirmar seu acesso no local selecionado. Repare que você só escolheu o campo **OTN**, mas o programa sabe que este campo do arquivo **VNOVO**, que abrimos no início de nossas operações com a impressão de arquivos múltiplos, está *comprometido* com sua sintaxe de programação de arquivos múltiplos, isto é, esta foi a forma pela qual o arquivo **VNOVO** foi transformado num arquivo múltiplo através da conexão dos arquivos **GRANA** e **PERFIL**. Por esta razão, vemos a sintaxe completa do campo programado, que inclui além do nome do campo selecionado, a seta mais o nome do arquivo de onde este campo foi retirado.

Após selecionar **OTN**, passe para a opção imediatamente abaixo, *Heading*, e pressione **ENTER**. Digite a palavra *valor* em letras minúsculas para sua posterior comparação. Vá para a próxima “página” de *Contents* com **PgDn** e repita o mesmo procedimento, desta vez escolhendo o campo **VENC** e digitando *vencimento*. Continue, trocando de página e selecionando o campo **STATUS** mais o cabeçalho em minúsculas: *situação*. Ao passar para as “páginas” seguintes, selecione os campos que faltam: **PRODUTO** (+ *produtos*) e **QTDE** (+ *quantidades*). Observe que esses dois últimos campos já não pertencem ao arquivo **GRANA.DBF** e sim ao arquivo **PERFIL.DBF**.

Ao terminar de preencher todos os campos, saia de *Columns* e salve tudo com *Exit* e *Save*.

Para gerar seu arquivo, acione a opção *Retrieve* e as subopções *Report*, *B: REVIEW.FRM*, *Execute the command* e, finalmente, *Y*, ou use os comandos digitados diretamente a partir do *prompt*, se preferir. Compare os resultados com a tela parcial que apresentamos na Figura 12.14.

Imprimindo Listagens de Programas (.PRG)

Para imprimir na íntegra seus programas que compõem o SuperMacro (**auto@.prg**, **REF.PRG** e **ETIQUETA.PRG**, simplesmente digite o comando **TYPE nome do programa TO PRINT**, a partir do *prompt*. Por exemplo, para imprimir **auto@**:

TYPE AUTO@.PRG TO PRINT ENTER

Page No. 1
06/29/88

RELATORIO REVIEW

valor	vencimento	situação	produtos	quantidade
** Cliente 001				
4.00	07/20/88	.T.	LOTUS T.A.	40
4.00	07/18/88	.T.	LOTUS T.A.	40
6.00	06/08/88	.F.	LOTUS T.A.	40
** Subtotal **				
14.00				120
 ** cliente 002				
15.00	07/20/88	.T.	WordLegal	60
** Subtotal **				
15.00				60
 ** cliente 003				
15.00	07/20/88	.T.	WordLegal	20
6.00	07/20/88	.T.	WordLegal	20
** Subtotal **				
21.00				40
 ** cliente 005				
4.00	06/30/88	.F.	MKT	10
** Subtotal **				
4.00				10
 ** cliente 007				
4.00	06/06/88	.F.	WordLegal	20
** Subtotal **				
4.00				20
 ** cliente 008				
4.00	06/22/88	.F.	MATFIN	80
** Subtotal **				
4.00				8
 ** cliente 009				
4.00	07/12/88	.T.	LOTUS T.A.	8
** Subtotal **				
4.00				8

Figura 12.14: Tela Parcial de REVIEW.FRM.

Para imprimir ETIQUETA.PRG, digite:

TYPE ETIQUETA.PRG TO PRINT ENTER

No próximo capítulo falaremos sobre projeto e desenvolvimento de programas. No Capítulo 14, Criação de Telas Personalizadas, aprenderemos a criar nossas próprias telas e me-

nus. No Capítulo 15, Conversões e Interfaces, falaremos sobre mais uma possibilidade especial de impressão. Abordaremos a interação com o SideKick, um software que permite a criação de uma tela específica dentro de sua tela do dBASE, para impressão e armazenamento de *qualquer* parte de sua tela.

Capítulo

13

**Projeto e Desenvolvimento
de Programas Incluindo
Técnicas de Depuração**

Neste capítulo abordaremos o projeto e o desenvolvimento de programas mais sofisticados e úteis. Nossa idéia original do “SuperMacro” `auto@.prg` do Capítulo 11 foi, na realidade, uma introdução (e um incentivo com nome sofisticado) à nossa verdadeira concepção do SuperMacro (daí a razão de sempre termos nos referido a ele entre aspas), muito embora essa seja uma concepção particular do autor para dar maior consistência à idéia, dos *arquivos de comando*, nome pelo qual são mais conhecidos.

Aproveitaremos também para apresentar algumas técnicas de depuração, úteis para a eventual correção e/ou otimização dos programas. Isto será feito na seção Técnicas de Depuração, mais adiante, após a análise de todos os programas.

Nossa proposta é construir um programa central (`central.prg`) para, por meio de um menu, acessar outros 10 programas utilitários que faremos na seqüência. A seguir, montaremos nosso primeiro utilitário, o programa `CASO1.PRG`, que operará com o arquivo `CLIEN- TES.DBF` e, quando chamado pelo programa central, poderá emitir relatórios na impressora ou exibir seu conteúdo somente na tela – tudo via decisão do controle central.

O programa `CASO2.PRG`, feito em seguida, será um pequeno programa para indicar duas opções de geração de etiquetas de endereçamento: impressão global – para operar com o arquivo `LIVRARIA.DBF` via `ETIQUETA.LBL` – e impressão específica – para operar com o arquivo `LIVRARIA.DBF` selecionando o registro desejado. Essas operações serão conseguidas por meio de outros dois programas auxiliares, `CASO2A` e `CASO2B`.

`CASO3.PRG`, a terceira opção do menu principal, operará com o arquivo `VENDAS.DBF` e apresentará duas opções: relatório impresso e exibição em tela. O programa seguinte, `CASO4.PRG`, a quarta opção do menu central, será um programa de atualização de todos os arquivos usados no sistema: `CLIENTES.DBF`, nossa agenda telefônica, `LIVRARIA.DBF`, nossa fonte de etiquetas de endereçamento e `VENDAS.DBF`, nosso arquivo de produtos, clientes e transações comerciais. Uma opção de saída para o menu principal também será prevista.

Mais três programas que serão chamados por `CASO4.PRG`: `NUM1`, `NUM2` e `NUM3.PRG`, respectivamente, por sua vez também apresentarão três opções: atualização – que permitirá a entrada automática no modo `APPEND` para atualização e posterior retorno ao submenu –, apagamento – que possibilitará a deleção de um registro selecionado pelo campo desejado – e uma alternativa de geração em tela ou relatório impresso. Uma opção de saída para o menu principal será apresentada como uma das alternativas desse menu.

Os programas `NUM1`, `NUM2` e `NUM3` executarão todos a mesma tarefa, em função do arquivo que tiver sendo solicitado por `CASO4`.

Para o caso de sua escolha ser o apagamento de um determinado registro em qualquer dos arquivos componentes do sistema, será acessado nosso programa final, o programa `Del@.prg`, que apagará por campo selecionado, qualquer registro indicado (não sem antes confirmá-lo).

PROGRAMANDO O SISTEMA

Programando o Menu Central

Vamos iniciar nossa programação com a construção do menu central de decisões. Sugerimos que você copie primeiro o programa para que nossa análise tenha maior sustentação.

```
***** Programa central.prg
***** Menu Central de Decisoes

set bell off
set status off
x = 0
do while x < > 5
clear
text
    MENU Principal
    (escolha um numero)

    1 = Agenda telefonica
    2 = Etiquetas postais
    3 = Relatorio de vendas
    4 = Atualizacao
    5 = Saida
endtext
@ 10,15 say 'Escolha um numero: ' get x picture '9' range 1,5
read
do case

    case x = 1
        do caso1
    case x = 2
        do caso2
    case x = 3
        do caso3
    case x = 4
        do caso4
endcase
enddo
set status on
set bell on
return
```

central.prg: Menu Central de Decisões.

Para início de conversa, temos que abrir um arquivo de programação para digitar e armazenar nosso programa. Esperamos que você faça isto com MODIFY COMMAND CENTRAL.

Com sua tela vazia copie então o programa central.prg.

Analisando os Procedimentos

O comando SET BELL OFF, no início do programa, diz ao dBASE para desligar o sinal sonoro do computador. Mais adiante, neste mesmo programa, exemplificaremos isto; o comando SET STATUS OFF diz ao dBASE para desligar a linha de *status* que a tela do dBASE apresenta na parte inferior do vídeo, que serve como referência dos procedimentos sendo utilizados no momento; $x = 0$ indica que uma variável que usaremos adiante foi inicializada corretamente, ou seja, ainda não tem valor definido pois ele só será definido quando teclarmos nossa opção, mais adiante; DO WHILE $X <> 5$ determina que adiante teremos até 5 opções para uma seleção numérica (de 1 a 5); CLEAR, como já sabemos, limpa a tela preparando-a para as próximas entradas; TEXT é o equivalente mais eficiente de “?” e “@”, para longas exposições de texto, ou seja, tudo o que for digitado entre TEXT e ENDTEXT será gerado na tela. É aí que definimos nosso primeiro menu, com as cinco opções principais.

Logo abaixo de ENDTEXT vemos o comando @ SAY, já discutido anteriormente. Aqui acrescentamos mais dois comandos ao nosso repertório: PICTURE e RANGE. PICTURE define o tipo de resposta que podemos inserir após a solicitação da escolha do número pelo programa. “9” indica que só poderemos inserir um número. “99” indicaria que poderíamos responder com dois números (nesse caso, se teclássemos somente um número, teríamos de teclar ENTER para dar seguimento ao programa. Com somente o espaço para um número, ao inserirmos esse número, o programa continua automaticamente, após o sinal sonoro característico de um campo que foi totalmente preenchido e já chegou ao fim. Esta é a razão pela qual desligamos o sinal sonoro no início do programa). Mais adiante mostraremos as combinações possíveis com PICTURE. RANGE define o maior e o menor número a ser inserido em resposta à solicitação do menu, ou seja, somente de 1 a 5 com direito a uma mensagem de erro, caso o número seja diferente.

READ completa o comando acima, fazendo com que o cursor se desloque para a posição de resposta.

DO CASE é o substituto mais ágil de IF. Ele apresenta todos os *casos* (CASE) definidos pelo menu, ou seja, caso x seja igual a 1 ($CASE\ X = 1$), então execute o programa *nome do programa* (DO CASO1). Repetimos esse mesmo procedimento até a última opção do menu (atualização).

A quinta opção, como já vimos, é a saída do programa e, portanto, não precisa ser prevista: ela já está automaticamente definida. Observe que demos os nomes de CASO1, CASO2, CASO3 e CASO4 aos programas chamados pelo menu. Você pode dar os nomes que quiser, contanto que não os misture com outros nomes posteriormente; o comando ENDCASE finaliza as operações definidas após DO CASE; ENDDO finaliza DO WHILE; SET STATUS ON liga a barra de status luminosa na parte inferior da tela do dBASE III Plus; SET BELL ON religa também o sinal sonoro emitido quando, por exemplo, preenchemos um campo completamente e, finalmente, RETURN retorna o controle para o *prompt* ou para outro programa do tipo .PRG.

Note que o formato de nosso programa é uma questão de estética/lógica, ou seja, todas as condições e/ou *loops* foram digitadas com um recuo de uma tabulação (pressionando a tecla Tab no modo Ins, com o cursor posicionado à esquerda, no início da linha de comando). É sempre mais fácil identificar determinadas partes do programa se as separarmos por setores lógicos. Em todos os 11 programas de nosso SuperMacro central.prg, usamos esse tipo de recuo para as cláusulas IF, DO WHILE e do CASE.

Ainda não podemos experimentar, por completo, nosso primeiro programa, tendo em vista que ainda não escrevemos os outros que fazem parte de todo o sistema, mas podemos testar algumas coisas. Por exemplo, vamos testar a seleção do menu. Saia da tela de central.prg com Ctrl-W e retorne ao *prompt*. A seguir, acione o comando:

DO CENTRAL ENTER

Tente colocar um número de 1 a 5. Repare que você recebe uma mensagem dizendo que o arquivo não existe:

```
File does not exist.
      ?
      do caso1
Called from - B:central.prg
Cancel, Ignore, or Suspend? (C, I or S)
```

A mensagem ocorreu porque ainda não temos nenhum programa chamado CASO1.

Tecla C e digite novamente:

DO CENTRAL ENTER

Tecla um número diferente de 1 a 5, 6, por exemplo. Agora a mensagem será:

```
RANGE is 1 to 5 (press SPACE)
```

que diz para você pressionar a barra de espaço e tentar de novo porque a faixa de números definida pelo programa (por nós) foi de 1 a 5 somente. Pressione a barra de espaço e, em seguida, tecla 5 para sair do menu diretamente para o *prompt*.

Chame sua listagem novamente com MODIFY COMMAND CENTRAL. Retire a linha de comando SET BELL OFF. Acione Ctrl-W e depois DO CENTRAL novamente. Tecla 5 novamente e você ouvirá o sinal sonoro.

Chame seu programa novamente com MODIFY COMMAND. Vá até o comando PICTURE na linha de @ SAY e coloque mais um "9" de modo que sua definição fique assim: PICTURE: "99". Salve tudo com Ctrl-W e, a seguir, digite:

DO CENTRAL ENTER

novamente, para então teclar o número 5. Note que agora nada acontece a não ser que teclamos ENTER. E foi para dar maior velocidade ao programa que definimos PICTURE com “9”, ou seja, somente um número, além de desligarmos o sinal sonoro para que possamos trabalhar com mais naturalidade.

Retorne ao programa e deixe tudo como estava antes do teste. Vamos para nosso segundo programa, que será o primeiro da série de aplicativos que interagirão com o menu central.

Refira-se ao Apêndice D caso você prefira adquirir o disquete com todos os programas deste livro já prontos para utilização com seu dBASE.

Programando CASO1.PRG

Copie o programa CASO1.PRG exatamente como você o vê no livro. Este programa será o programa escolhido pelo programa principal, central.prg, caso nossa escolha seja o número 1.

```
***** Programa CASO1.PRG
***** Opera com o arquivo CLIENTES.DBF

set talk off
y = space(1)
clear
@ 10,15 say 'Escolha um numero. Relatorio = 1. Tela = 2:' get y
read
if y = '2'
    use clientes
    clear
    list
    wait 'Pressione qualquer tecla para retornar ao menu anterior' to tecla
clear
endif
if y = '1'
    use clientes
    clear
    list to print
    wait 'Pressione qualquer tecla para retornar ao menu anterior' to tecla
clear
endif
set talk on
return
```

CASO1.PRG: Opera com o arquivo clientes.dbf.

Analizando os Procedimentos

Neste momento de sua leitura deste livro, supomos – corretamente, esperamos –, que você já esteja dominando as técnicas já discutidas no Capítulo 11 e no programa central.prg que

acabamos de ver. Assim, nossos comentários serão, doravante, um pouco mais breves.

Usamos SET TALK OFF para definir que não queremos que os números dos registros apareçam em nossa tela/relatório impresso (em caso de dúvida execute o programa sem esta linha, depois coloque-a de volta); *y = space(1)* funciona como PICTURE e define apenas um espaço para preenchimento da opção adiante; CLEAR limpa a tela; a linha @ SAY pede que escolhamos um número, definido como 1 ou 2; READ coloca o cursor na posição da resposta; a cláusula IF abaixo diz: “se y for igual a 2, abra o arquivo CLIENTES.DBF, limpe a tela e liste o arquivo na tela. O comando WAIT faz com que o programa aguarde pelo pressionamento de uma tecla para retornar ao menu anterior (que, no final dos 11 programas, poderá ser o menu central.prg ou o menu de atualização); novamente, CLEAR limpa a tela e ENDIF finaliza a primeira condição IF; a seguir, outra condição IF (1) determina que o arquivo CLIENTES.DBF será aberto e que desejamos uma listagem impressa de seu conteúdo; marcamos o final do programa com mais um comando CLEAR para limpar a tela, ENDIF para terminar mais uma condição IF, SET TALK ON para reativar os números dos registros e RETURN para retornar o controle para o programa chamador.

Programando CASO2.PRG

O programa CASO2.PRG, a segunda opção de nosso menu principal em central.prg, é um pequeno programa de geração de etiquetas de endereçamento baseado no arquivo fonte LIVRARIA.DBF. Copie-o agora e depois vamos aos comentários sobre sua estrutura.

```
***** Programa CASO2.PRG
***** Submenu de etiquetas de endereçamento

R = space(1)
clear
@ 10,15 say 'Tipo de impressao: Global = 1. Especifica = 2: ' get R
read
if R = '2'
    do caso2A
endif
if R = '1'
    do caso2B
endif
return
```

CASO 2.PRG: Programa Chamador de CASO2A e CASO2B.

O programa CASO2.PRG produz etiquetas de endereçamento de duas formas. Se o usuário optar pela opção 1, Global, então o subprograma CASO2B será acionado; se a escolha for gerar etiquetas selecionadas, então o programa chamará o subprograma CASO2A.

Passemos então para CASO2A.

Programando CASO2A.PRG: Corrigindo ETIQUETA.PRG

```

***** Programa CASO2A.PRG (Selecoes especificas)
***** Opera com o arquivo LIVRARIA.DBF

set talk off
clear
use livraria
accept 'Deseja pesquisar em qual campo?: ' to C
accept 'Procurar qual &C: ' to COND
accept 'Deseja imprimir? (S/N): ' to R
clear
if upper(R) = 'S'
    set print on
endif
go top
do while .not. eof( )
if upper(&C) = upper('&COND')
    ? nome
    ? endereco
    ? trim(cidade)+'-' +estado
    ? cep
    ?
    ?
    ?
endif
skip
enddo
set print off
wait 'Pressione qualquer tecla para retornar ao menu anterior' to tecla
clear
set talk on
return

```

CASO2A.PRG: Opera com o Arquivo LIVRARIA.DBF.

O programa CASO2A tem algumas semelhanças com nosso primeiro programa de geração de etiquetas de endereçamento. ETIQUETA.PRG, que criamos juntamente com o conceito de “SuperMacro” no Capítulo 11. Vamos aproveitar agora e corrigir alguns pequenos defeitos de ETIQUETA.PRG, enquanto comentamos CASO2A.

Analizando os Procedimentos

O início de ambos os arquivos é quase idêntico, não fosse uma diferença estética que pode vir a atrapalhar: o programa ETIQUETA.PRG não traz no seu título o nome do programa, dificultando um pouco uma documentação que pretende ser eficiente. Logo abaixo do comando DO WHILE, no programa ETIQUETA.PRG lê-se “*if &C = “&COND”*”, o que significa “se o conteúdo do campo nome do campo for igual ao nome digitado pelo usuário, então processe isso...”. Note que, com ETIQUETA.PRG, só nos foi possível escrever o registro desejado *exatamente* como ele se encontrava no arquivo LIVRARIA.DBF. Re-

pare agora a mesma linha modificada no programa CASO2A. Note que ela diz: “*if upper(&C) = upper('&COND')*”, que quer dizer “se o conteúdo do campo *nome do campo* quando convertido para letras maiúsculas for igual ao nome digitado pelo usuário também convertido para maiúsculas, então processe isso...”. Ou seja, agora podemos digitar nossas seleções da forma que desejarmos, como “NOME”, “nome”, “NoMe”, ou “nOmE”. Note também que no programa ETIQUETA.PRG, programamos uma pequena rotina no final para optar ou não pelo retorno ao programa principal, *auto@.prg*. Aqui isto não acontece, pois não é necessário, visto que ele retorna automaticamente ao programa chamador se incluirmos a instrução RETURN no final. (Qualquer dúvida recorra à análise dos procedimentos de estruturação desse programa no Capítulo 11.)

Programando CASO2B.PRG

Copie este pequeno programa que operará com o arquivo LIVRARIA.DBF via ENDEREÇA.LBL – o arquivo especial de etiquetas de endereçamento construído com o comando MODIFY LABEL, no capítulo anterior.

```
***** Programa CASO2B.PRG (Todas as etiquetas do arquivo)
***** Opera com o arquivo LIVRARIA.DBF
***** (extraído de ENDERECA.LBL)

use livraria
set talk off
label form endereca to print
wait 'Pressione qualquer tecla para retornar ao menu anterior' to tecla
set talk on
return
```

CASO2B.PRG: Opera com o Arquivo LIVRARIA.DBF via ENDERECA.LBL.

Analisando os Procedimentos

Se escolhida a opção 1, Global, da opção 2 do menu principal, seu programa gerará diretamente, sem perguntas, todas as etiquetas do arquivo.

Programando CASO3.PRG

Copie CASO3.PRG.

```
***** Programa CASO3.PRG
***** Opera com o arquivo VENDAS.DBF

use vendas
set talk off
y = space(1)
clear
```



```

@ 10,15 say 'Escolha um numero. Relatorio = 1. Tela = 2: ' get y
read
if y = '2'
    list
    wait 'Pressione qualquer tecla para retornar ao menu anterior' to tecla
clear
endif
if y = '1'
    report form vendas to print
    wait 'Pressione qualquer tecla para retornar ao menu anterior' to tecla
clear
endif
set talk on
return

```

CASO3.PRG: Opera com o Arquivo VENDAS.DBF.

Analizando os Procedimentos

A configuração deste arquivo é semelhante à do arquivo CASO1 já visto. A diferença é que aqui usamos o comando REPORT FORM para gerar o relatório (preparado no capítulo anterior) em vez de LIST TO PRINT, que gera um relatório sem editoração, diretamente da tela do arquivo VENDAS.DBF.

Programando CASO4.PRG

CASO4.PRG é um programa chamado pela escolha do número 4 no menu principal, que, por sua vez, também chama outros três programas que serão analisados em seguida. Copie CASO4 exatamente como no exemplo.

```

***** Programa CASO4.PRG
***** Atualizacao de arquivos

x = 0
do while x <> 4
clear
text
    Atualizar:
    1 = Agenda
    2 = Etiquetas
    3 = Vendas
    4 = Saida
endtext
@ 10,15 say 'Escolha um numero: ' get x picture '9' range 1,4
read
do case
    case x = 1
        do num1
    case x = 2
        do num2

```

```

    case x = 3
      do num3
    endcase
  enddo
return

```

CASO4.PRG: Atualização de Arquivos.

Analizando os Procedimentos

Este é mais um programa-menu, para decisão sobre qual arquivo atualizar. Podemos atualizar nossa agenda telefônica (CLIENTES.DBF), nosso arquivo-fonte de etiquetas de endereçamento (LIVRARIA.DBF) e nosso arquivo comercial de vendas (VENDAS.DBF). Tal qual o programa central, esse arquivo também apresenta suas opções (4 – incluindo a opção de saída) e três programas próprios chamados NUM1, NUM2 e NUM3, respectivamente.

Programando NUM1.PRG

Copie o programa NUM1.PRG exatamente como você o vê no livro.

```

***** Programa NUM1.PRG – Chamado de CASO4.PRG
***** Atualiza arquivo CLIENTES.DBF

use clientes
x = 0
do while x < > 4
clear
text
  Seleccione:
  1 = Atualiza (use PgUp PgDn. Ctrl-END para sair)
  2 = Apaga registros (por campos)
  3 = Gera: tela
      relatorio impresso
  4 = Retorna ao Menu Principal
endtext
@ 10,15 say 'Escolha um numero: ' get x picture '9' range 1,4
read
do case
  case x = 1
    append
  case x = 2
    do del@
  case x = 3
    do caso1
  endcase
enddo
return

```

NUM1.PRG: Chamado de CASO4.PRG: atualiza CLIENTES.DBF.

Este programa é a base para os outros dois que serão feitos a seguir. A única diferença é que cada um deles opera com seu respectivo arquivo, ou seja, NUM1 começa abrindo o arquivo CLIENTES.DBF, NUM2 inicia abrindo o arquivo LIVRARIA.DBF e NUM3 começa abrindo VENDAS.DBF. Vamos a uma pequena análise.

Analizando os Procedimentos

O programa NUM1.PRG atua como menu de atualização do arquivo CLIENTES.DBF. Com ele, podemos: 1) atualizar o arquivo; 2) apagar os registros indicados; 3) gerar uma tela ou relatório impresso (via CASO1) de seu conteúdo e, finalmente, retornar ao menu principal, se preferirmos. Para o primeiro caso, NUM1 recorre ao comando APPEND; para o segundo caso, NUM1 acessa o programa del@.prg, o último programa do sistema, usado para apagar registros selecionados de todos os arquivos do sistema; 4) retornar ao menu principal.

Programando NUM2.PRG e NUM3.PRG

Copie os arquivos NUM2.PRG e NUM3.PRG como você os vê no livro.

***** Programa NUM2.PRG – Chamado de CASO4.PRG

***** Atualiza arquivo LIVRARIA.DBF

```

use livraria
x = 0
do while x < > 4
clear
text
  Seleccione:
  1 = Atualiza (use PgUp PgDn. Ctrl-END para sair)
  2 = Apaga registros (por campos)
  3 = Gera: tela
      relatorio impresso
  4 = Retorna ao Menu Principal
endtext
@ 10,15 say 'Escolha um numero: ' get x picture '9' range 1,4
read
do case
  case x = 1
    append
  case x = 2
    do del@
  case x = 3
    do caso2
endcase
enddo
return

```

NUM2.PRG: Chamado de CASO4.PRG: Atualiza arquivo LIVRARIA.DBF.

```

***** Programa NUM3.PRG – Chamado de CASO4.PRG
***** Atualiza arquivo VENDAS.DBF

use vendas
x = 0
do while x < > 4
clear
text
    Seleccione:
    1 = Atualiza (PgUp PgDn. Ctrl-END para sair)
    2 = Apaga registros (por campos)
    3 = Gera: tela
        relatorio impresso
    4 = Retorna ao Menu Principal
endtext
@ 10,15 say 'Escolha um numero: ' get x picture '9' range 1,4
read
do case
    case x = 1
        append
    case x = 2
        do del@
    case x = 3
        do caso3
endcase
enddo
return

```

NUM3.PRG: Chamado de CASO4.PRG: Atualiza arquivo VENDAS.DBF.

Analizando os Procedimentos

Esses dois programas completam a série de programas do tipo NUM que, por sua vez, complementam CASO4.PRG. Ambos atualizam seus respectivos arquivos; ambos acessam o mesmo programa Del@.prg sendo que, na opção 3, NUM2 acessa CASO2 e NUM3 acessa CASO3, formando assim, um *loop* controlado pelo comando central (central.prg).

Programando Del@.prg

Copie nosso último programa deste capítulo, Del@.prg, que apaga registros selecionados de qualquer um dos arquivos-fonte considerados até agora.

```

***** Programa Del@.prg. (Chamado de NUM1, NUM2 e NUM3 ->CASO4.PRG)
***** Apaga registros

clear
accept 'Deseja pesquisar em qual campo?: ' to C
accept 'Procurar qual &C: ' to COND
accept 'Deseja ver o registro? (S/N): ' to R

```

```

clear
if upper(R) = 'S'
    list for upper(&C) = upper(COND)
endif
accept 'Quer apaga-lo? (S/N): ' to A
if upper(A) = 'S'
    clear
    delete for upper(&C) = upper(COND)
endif
list
wait '(...ENTER continua)' to tecla
x = 0
clear
@ 10,15 say 'Confirmar seu apagamento? Sim = 1. Nao = 2.: ' get x picture '9' range 1,2
read
do case
    case x = 1
        pack
        clear
        list
        wait 'Sua tela atual. (... ENTER retorna ao Submenu)'
    case x = 2
        recall all
        list
        wait 'Sua tela atual. (... Enter retorna ao Submenu)'
return

```

Del@.prg: Chamado de NUM1, NUM2 e NUM3: Apaga registros.

Analizando os Procedimentos

O programa Del@.prg, chamado da série de programas NUM, que por sua vez é chamada de CASO4.PRG, apaga qualquer registro determinado pelo usuário, de qualquer dos arquivos que fazem parte de nosso sistema: CLIENTES.DBF, VENDAS.DBF e LIVRARIA.DBF.

Note que não “amarramos” este programa, colocando, por exemplo, uma abertura de arquivo do tipo *USE nome de arquivo*, no início. Isto deixa o programa livre para ser chamado por qualquer dos programas NUM, que já deverá ter um dos arquivos abertos. Se não fizéssemos isso, teríamos fatalmente que incluir mais dois programas Del@.prg.

Analise a sintaxe de cada um dos programas de nosso sistema para familiarizar-se com eles, comparando-os entre si.

TÉCNICAS DE DEPURAÇÃO

Aqui – e agora –, é um bom momento para discutirmos algumas técnicas de depuração de programas. Entenda-se por depuração, o desempenho correto de um programa via tes-

tes/pesquisa/correção e/ou otimização do programa. Começaremos por algumas dicas importantes e mais ou menos aleatórias, no que se refere a nossos programas recentes.

Perceba que não é preciso incluir o comando SET TALK OFF/ON, quando se trata de um programa exclusivamente de menu, ou seja, como nada será impresso no próprio menu e sim num programa chamado por ele, é melhor deixar para o programa chamado a inclusão do comando.

Você deve ter notado que, durante suas interações com todo o sistema, muitas vezes, ao retornar de um programa chamado pelo programa central, você encontrou o número da última opção na posição de seleção do menu. Quando isto acontece – e você ainda desejar voltar à mesma opção, simplesmente pressione ENTER sem teclar novamente o número da seleção.

Caso você esteja no menu principal, em central.prg e, ao selecionar um número, você se arrependa e desista de sua seleção, simplesmente pressione ESC para retornar ao mesmo menu.

Alguns autores preferem incluir em seus arquivos de comando (o que aqui rebatizamos de SuperMacros) a indexação do arquivo sendo acessado. Aqui não fizemos isso por acharmos que o processo de indexação do dBASE III Plus é muito lento para fazer parte de um programa como este (o dBASE IV acelerará este processo). Aconselhamos você a tentar um arquivo separado com a indexação de *todos* os campos que porventura sejam necessários em suas futuras interações, já prontos para acesso (um 12º programa). Isto certamente tornaria o processo muito mais rápido e consistente, visto que você poderia acessar qualquer arquivo/campo indexado e, principalmente, não teria que apagar esse arquivo no final do programa, como é recomendado no caso de haver a indexação dentro do programa.

Outro detalhe que você pode ter notado, foi a ausência de mensagens do tipo REM de BASIC (os “famosos” asteriscos usados para mensagens e lembretes que não são impressos quando o programa é aplicado), em todos os nossos programas, com exceção dos títulos. Isto acontece porque, ao contrário do que a maioria dos autores advoga, achamos mais inteligível um programa “limpo”, mais rápido de ser feito e, quando comentado numa seção específica, como fizemos todo o tempo em Analisando os Procedimentos, mais fácil de ser compreendido.

Outra coisa para se observar é o tamanho de suas linhas de comando. Pode-se terminar uma linha de comando em baixo, não havendo necessidade de que ela termine na própria linha original. (Porém, alguns compiladores não aceitarão esse tipo de linha quando não houver um final lógico).

Ao chamar a alternativa de seleção de campos específicos, para impressão de etiquetas, por exemplo, pode-se desistir da operação e visualizar todos os campos do arquivo-fonte, simplesmente pressionando-se ENTER para a segunda pergunta (...qual estado?).

Para verificar o andamento de um programa, você pode dispor de algumas “ferramentas” de trabalho muito eficientes. Por exemplo, para acompanhar toda a programação de um determinado programa do tipo .PRG, pode-se, a partir do *prompt*, digitar:

SET ECHO ON ENTER

e a seguir,

DO nome do programa ENTER

Agora, todo o programa rodará juntamente com sua respectiva programação e você poderá acompanhar o desenvolvimento de sua lógica.

Ao terminar de usar ECHO ON, desligue a função via

SET ECHO OFF ENTER

Este comando é semelhante ao comando TRON e TROFF do BASIC, muito eficiente por sinal.

Para diminuir ainda mais a velocidade de sua averiguação, acione o comando

SET STEP ON ENTER

para acompanhar seu programa, linha-a-linha, com ENTER. Em contrapartida, o comando:

SET STEP OFF ENTER

desliga essa atividade.

Uma opção complementar de STEP e ECHO, é o comando:

SET DEBUG ON ENTER

que envia toda a saída de ECHO ou de STEP para a impressora. Novamente,

SET DEBUG OFF ENTER

desativará o comando.

Lembre-se também que você pode prever o acionamento desses comandos dentro de um arquivo de comandos, ou seja, de um programa do tipo .PRG, contanto que seja bem aplicado nas áreas de real necessidade (por exemplo, não é necessário um comando do tipo SET ECHO ON antes de um comando do tipo USE LIVRARIA que inserimos no programa CASO2A.PRG).

A Memória dos Comandos do Teclado

Pode-se ativar a memória de seus últimos comandos emitidos no teclado via

DISPLAY HISTORY ENTER

diretamente do *prompt*. Com este comando, pode-se recuperar os últimos 20 comandos acionados pelo usuário no teclado. Pode-se também, para um acompanhamento mais extenso, alterar o *default* de 20 para 30, ou mais, via comandos:

SET HISTORY TO 30 ENTER

para passar o *default* para 30;

SET DOHISTORY ON ENTER

para incluir seu “histórico” dos últimos 30 comandos quando você acionar, novamente:

DISPLAY HISTORY ENTER

ou

LIST HISTORY ENTER

ou ainda:

DISPLAY HISTORY TO PRINT ENTER

ou

LIST HISTORY TO PRINT ENTER

para gerar seus resultados na impressora.

Após terminar com sua verificação, volte para o padrão de 20 com:

SET DOHISTORY OFF ENTER

Experimente alguns comandos com o complemento...TO PRINT, como DISPLAY STRUCTURE TO PRINT, LIST TO PRINT etc.

Um importante “histórico” de comandos pode ser verificado, tanto para atividades do menu (não confunda com *no* menu) como para atividades do *prompt*, isto é, seu teclado armazena automaticamente os últimos 20 comandos digitados, na tecla com a seta para cima (e para baixo, no caso de retorno). Experimente agora! Tente visualizar seus últimos 20

comandos pressionando a tecla com a seta para cima várias vezes. Retorne para os comandos mais recentes com a tecla com a seta para baixo. Este é um importante dispositivo de trabalho, visto que com ele podemos também retroceder a qualquer hora, pressionar ENTER sobre um comando acionado anteriormente e obter a resposta imediata, sem ter de digitar o tempo todo (por isso, selecione bem seus comandos do *prompt*, não os desperdiçando, para que seu “histórico” de comandos esteja sempre entre os 20 automaticamente guardados na tecla com a seta para cima). Veja também SET TYPEHEAD TO, na Lista de Comandos Set adiante.

SET ALTERNATE também pode ser usado para compor um “histórico” de comandos e entradas na tela em geral. Sua sintaxe é:

SET ALTERNATE TO *nome do arquivo* **ENTER**

para criar um arquivo no qual serão guardados todos os comandos emitidos durante a ativação do comando, complementado por

SET ALTERNATE ON **ENTER**

para ativar o comando.

Com SET ALTERNATE ON, tudo o que você digitar será armazenado em um arquivo especial com a extensão .TXT, o que significa que, para visualizar este relatório de comandos, ou seja, seu “histórico” de comandos, você terá de acessar seu arquivo via processador de texto (o autor trabalhou neste livro com o excelente *WordStar Professional Release 4**. Para isso, simplesmente saia o dBASE após terminar de gravar seu arquivo (você não precisa fazer nada, continue trabalhando e o dBASE armazenará suas interações neste arquivo específico, criado com SET ALTERNATE TO *nome do arquivo* seguido de SET ALTERNATE ON. Enquanto o comando estiver ON, tudo será guardado. Para deixar de armazenar comandos, desligue ALTERNATE com:

SET ALTERNATE OFF **ENTER**

e tudo voltará a ser como antes, exceto pelo fato de que agora seu trabalho já não mais será gravado.

Para “ler” seu arquivo/histórico de comandos, como já dissemos antes, saia do dBASE, deixe o disco de trabalho (aquele que contém o arquivo criado com SET ALTERNATE) no acionador B e coloque seu disquete do WordStar no acionador A. A seguir, tecle, **WS**. Quando o menu de opções de seu processador de texto (qualquer um) aparecer, chame seu arquivo gravado com SET ALTERNATE pelo seu nome completo. Isso é tudo! (Não é fácil?)

*N.E.: Consulte-nos sobre nossa publicação deste livro para 1988/89.

Você também pode gravá-lo num disco de trabalho exclusivo de seus arquivos de texto. Para isso, na hora de salvá-lo novamente, salve-o em seu disquete de trabalho usado para suas interações com seu processador de texto.

O comando

SUSPEND

pode ser usado para suspender um programa do dBASE. Use-o para restaurar algum comando ou linha de programa mal feita, sempre que receber a mensagem.

Cancel, Ignore ou Suspend? (C, I, or S)

em decorrência de alguma parada involuntária de seu programa. Para continuar novamente o programa, depois de encontrado e reparado o erro, digite:

RESUME

Seu programa deverá continuar a partir do ponto em que parou.

Mais Depuração com Comandos SET

Agora vamos relacionar mais algumas funções do comando SET seguido de seu complemento que irá determinar a ação.

Pode-se usar o comando SET CARRY ON para auxiliar em operações de APPEND. Por exemplo, quando acionamos APPEND, recebemos uma tela vazia de atualização do arquivo chamado, que vem a ser o próximo registro após o final do arquivo. SET CARRY, entretanto, sempre que ativado, traz para dentro desse registro vazio o último registro de seu arquivo (se acionarmos PgDn, esse registro se repetirá nos registros subseqüentes o mesmo número de vezes que PgDn for pressionada). Este comando pode ser utilizado para que o usuário saiba imediatamente qual é seu último registro, para facilitar o processo de atualização de um determinado arquivo.

Como de hábito, para desativar o comando SET CARRY ON, use SET CARRY OFF.

Outro comando do tipo SET que analisaremos aqui (no final deste capítulo relacionaremos todos os comandos SET, numa tabela específica) é SET COLOR. Este comando permite que programemos as cores de nosso vídeo – ou que mudemos sua tonalidade, no caso de um monitor monocromático. Por exemplo, acione o comando:

SET COLOR TO R/W, B/G ENTER

e veja o resultado. (Use a tabela de referência a seguir, para escolher seu próprio padrão.)

Para retornar ao padrão monocromático, acione o comando:

SET COLOR TO W/N, N/W ENTER

Pode-se usar várias combinações de uma só vez, para isso consulte a Tabela de Código de Cores.

Cor	Código
azul	B
azul-escuro	BG
branco	W
magenta	BR
marron	GR
preto	N
transparente	X

Tabela de Código de Cores

Pode-se também selecionar cores via menu. O comando para isso é:

SET

Para entrar na tela dos comandos SET. A seguir, vá para a opção **Screen**. Use as seleções disponíveis com a seta para cima e para baixo, para definir as cores desejadas.

Já aplicamos o comando SET FUNCTION anteriormente, via menu. Basta digitar SET ENTER para que o menu de funções do tipo SET apareça para possíveis modificações. Mas se você preferir digitar diretamente do *prompt*, acione o comando:

SET FUNCTION 4 TO 'BROWSE'; ENTER

para, por exemplo, mudar a atribuição da tecla de função 4 para o comando BROWSE. Não se esqueça de colocar o ponto-e-vírgula, pois isto significa que seu comando será seguido também de ENTER, automaticamente.

Como Abreviar Comandos

Você também pode abreviar comandos, para até quatro letras, para economizar algum tempo na sua digitação. Por exemplo, para digitar DISPLAY STRUCTURE, bastaria digitar DISP STRU.

O mesmo acontece com os outros comandos. Pratique um pouco agora com uma variedade de comandos (lembre-se de que seus comandos ficarão armazenados na seta para cima. Não será preciso digitá-los novamente, caso precise de alguns dos que você usou para praticar).

Usando Substrings

A função “\$” (um cifrão), quando colocada no início da especificação de um campo que será pesquisado, retornará o registro solicitado pelo conteúdo desse campo que tenha, como parte integrante, qualquer palavra que o componha. Por exemplo, o comando

LIST FOR ‘Aguias’ \$ENDERECO ENTER

quando acionado para o arquivo CLIENTES.DBF, achará o registro pedido somente via ocorrência de uma só palavra pertencente ao total do endereço (outras funções poderão ser consultadas no final deste livro).

LISTA DE COMANDOS SET

A seguir relacionaremos todos os comandos do tipo SET do dBASE III Plus.

SET ALTERNATE TO

Define um arquivo especial para armazenamento no formato .TXT.

SET ALTERNATE ON/OFF

Armazena o que tiver sido bloqueado com SET ALTERNATE TO *nome do arquivo*.

SET BELL ON/OFF

Ativa/desativa o sinal sonoro.

SET CARRY ON/OFF

Traz para o próximo registro vazio, o último registro de uma tela APPEND.

SET CATALOG TO

Define um arquivo-catálogo (um arquivo do tipo *diretório*, que, por sua vez, conterá todos os arquivos que forem guardados pelo usuário no catálogo).

SET CATALOG ON/OFF

Ativa/desativa um catálogo de arquivos, definido pelo comando SET CATALOG TO *nome do arquivo*.

SET CENTURY ON/OFF

Ativa/desativa a exibição do século atual em uma data.

SET COLOR TO

Ajusta os atributos de cores de um monitor colorido – e, com algumas limitações –, também de um monitor monocromático.

SET COLOR ON/OFF

Ativa/desativa os modos de cores e monocromático.

SET CONFIRM ON/OFF

Avanço do cursor, sem necessidade de se pressionar ENTER, após o preenchimento de um campo.

SET CONSOLE ON/OFF

Ativa/desativa a visualização de dados na tela.

SET DATE

Ativa os formatos de data definidos pelo usuário. São eles: AMERICAN, ANSI, BRITISH, FRENCH, GERMAN, ITALIAN. (No formato ANSI o ano é o primeiro; no formato GERMAN o dia vem primeiro; no formato AMERICAN o mês aparece primeiro.)

SET DEBUG ON/OFF

Direciona a saída do comando SET ECHO ON para a impressora, para fins de rastreamento de um programa.

SET DECIMALS TO

Especifica o número de casas decimais de uma operação de cálculo.

SET DEFAULT TO

Especifica o acionador (*drive*) a ser usado para trabalho.

SET DELETED ON/OFF

Ignora, quando em OFF (não exibe) registros marcados para deleção com o asterisco.

SET DELIMITER TO

Especifica delimitadores de campo para exibição na tela.

SET DELIMITER ON/OFF

Determina o modo de entrada dos campos exibidos, como vídeo inverso, etc..

SET DEVICE TO

Determina a saída de comandos formatados.

SET DOHISTORY ON/OFF

Armazena um “histórico” dos comandos já utilizados pelo usuário.

SET ECHO ON/OFF

Determina se os comandos executados por um programa serão exibidos para rastreamento.

SET ESCAPE ON/OFF

Ativa/desativa a tecla Esc.

SET EXACT ON/OFF

Determina como os *strings* serão comparados: se EXACT estiver ON, então o conteúdo dos campos fornecidos pelo usuário deverá conter o tipo exato de apresentação do conteúdo dos campos do programa.

SET FIELD TO

Determina o acesso aos campos por comandos diretos.

SET FIELDS ON/OFF

Ativa/desativa uma lista de campos especificados pelo comando SET FIELD TO.

SET FILTER TO

Exibe somente os registros que atendam a uma condição especificada. Esta condição pode ser somente um determinado conteúdo de um campo.

SET FIXED ON/OFF

Ativa/desativa o comando SET DECIMAL.

SET FORMAT TO

Seleciona um arquivo de formatação.

SET FUNCTION TO

Programa as teclas de função.

SET HEADING ON/OFF

Ativa/desativa a geração de cabeçalhos e/ou títulos.

SET HELP ON/OFF

Ativa/desativa a mensagem *Do you want some help? (Y/N)* que ocorre quando o dBASE percebe que você cometeu algum engano.

SET HISTORY TO

Determina o número de linhas a ser armazenado em um “histórico” de linhas.

SET HISTORY ON/OFF

Ativa/desativa um “histórico” de comandos. O *default* é de 20 comandos, mas pode-se alterá-lo para mais.

SET INDEX TO

Seleciona e abre um arquivo indexado especificado.

SET INTENSITY ON/OFF

Ativa/desativa o vídeo inverso.

SET MARGIN TO

Controla a margem esquerda da impressora. Seu *default* é zero.

SET MEMOWIDTH TO

Especifica o tamanho da coluna de um campo memo.

SET MENU ON/OFF

Ativa/desativa a exibição dos menus.

SET MESSAGE TO

Determina uma mensagem, definida pelo usuário, na parte inferior da tela.

SET ORDER TO

Cria um arquivo indexado sem precisar fechar ou abrir arquivos.

SET PATH TO

Seleciona uma via de acesso a um arquivo.

SET PRINT ON/OFF

Envia a saída diretamente para a impressora (ON) ou somente para a tela (OFF).

SET PRINTER TO

Redireciona a impressão para o dispositivo especificado pelo DOS.

SET PROCEDURE TO

Seleciona (cria) um arquivo de *procedures*.

SET RELATION TO

Conecta dois bancos de dados por um campo em comum.

SET SAFETY ON/OFF

Ativa/desativa as mensagens de alerta para a sobregravação de arquivos.

SET SCOREBOARD ON/OFF

Ativa/desativa a exibição de informações da linha de *status*.

SET STATUS ON/OFF

Ativa/desativa a exibição da barra de *status* na parte inferior da tela.

SET STEP ON/OFF

Ativa/desativa o rastreamento de um programa linha-a-linha.

SET TALK ON/OFF

Ativa/desativa a exibição de resultados de cálculos e comandos na tela (ativa/desativa a exibição dos números de registro).

SET TITLE ON/OFF

Ativa/desativa o *prompt* para a descrição de um arquivo se o catálogo estiver ativo.

SET TYPEAHEAD TO

Controla o número de caracteres no *buffer* do teclado. O *default* é 20.

SET UNIQUE ON/OFF

Ativa/desativa a impressão de registros não duplicados. (Usado com INDEX.)

SET VIEW TO

Abre um arquivo do tipo .VUE.

AUMENTANDO A VELOCIDADE COM PROCEDURES

O uso de *procedures* (procedimentos) é uma alternativa interessante para se conseguir uma velocidade maior de resposta para programas que se utilizam de diversas rotinas ou subprogramas, com os 11 programas que compõem nosso SuperMacro central.prg. Com essa ferramenta de programação, você pode armazenar todos os procedimentos (os subprogramas) de uma só vez na memória, de modo que o acionador não precise ser utilizado com tanta frequência para as consultas a cada programa.

Devemos lembrar, porém, que nossa rede de programas central.prg já possui uma velocidade acentuada! Entretanto, o uso de *procedures* não deve ser esquecido.

A grande diferença entre nossos programas do SuperMacro central.prg e os mesmos programas escritos com *procedures* é que, com nossos 11 programas, o computador tem de chamar cada um, à medida que é solicitado, ao mesmo tempo que o usuário tem de armazenar cada um dos programas separadamente, com Ctrl-W no modo MODIFY COMMAND, por nome (um nome para cada arquivo de comando).

No caso de um arquivo de comando por *procedures*, deve-se armazenar todas as rotinas (no caso anterior, os 10 programas que compõem a rede controlada por central.prg – que será o programa chamador dos *procedures* – que não é conectado à rede) num único programa, separado apenas pela indicação do início (PROCEDURE *nome do arquivo*) e do fim (RETURN) de cada rotina ou programa. Como exemplo, vamos usar o SuperMacro central.prg e seus 10 programas, para juntá-los num único programa e, via *procedures*, renomeá-lo como um só programa, além do programa central.

Para juntar os 10 programas, iniciando com CASO1.PRG e terminando com Del@.prg, na ordem hierárquica, o autor utilizou o processador de texto *WordStar Professional Release 4* da seguinte forma que sugerimos você copie:

1) Retire o símbolo “@” do programa Del@.prg, pois ele não pode ser usado com *procedures* (nem para compilação!). Não se esqueça de retirá-lo também dos programas onde ele aparece como referência.

2) Acesse `central.prg` e escreva no início de sua listagem, logo acima do comando `SET BELL OFF`, de cima para baixo: `SET PROCEDURE TO AUTOPROC` (*autoproc* é o nome do arquivo de procedimentos que criaremos a seguir); em baixo deste comando, insira: `PROCEDURE CENTRAL`, que é o nome de nosso primeiro procedimento (os procedimentos que interessam mais são os de `autoproc.prg` que será criado daqui a pouco).

3) Certifique-se de que todos os programas seguintes (de `CASO1` até `Del.prg`) tenham a instrução `RETURN` no final.

4) Retire os cabeçalhos de todos os programas restantes (isto só será necessário porque seu arquivo ficará quase no limite de tamanho possível para esse tipo de arquivo) e coloque o comando `PROCEDURE nome do programa` no início de cada um dos programas do grupo de 10 (entende-se por *nome do programa* os nomes de cada um deles, exatamente como os chamamos).

5) Saia do dBASE (antes certifique-se de que todos os arquivos estão a salvo em uma cópia de reserva, para o caso de alguma perda acidental) com `QUIT`, mas deixe seu disquete de trabalho no acionador B.

6) Entre no WordStar (ou em seu processador de texto – caso seja outro). Lembre-se de que nosso exemplo foi executado com um *WordStar Professional Release 4*, que é o que usaremos também para nosso exemplo agora (qualquer dúvida, informe-se com seu fornecedor de *software* sobre a equivalência de comandos entre uma versão e outra do WordStar). Acione o comando `WS` diretamente do *prompt* do DOS. Quando o WordStar apresentar seu menu de abertura, tecle a opção “N”, de *nondocument*. A seguir, chame o arquivo `CASO1.PRG` do acionador B. Quando seu arquivo aparecer na tela (o programa `CASO1.PRG` no disquete de trabalho do dBASE III plus), já modificado com o título `PROCEDURE CASO1` no lugar dos cabeçalhos, leve o cursor até o final do programa e salte umas duas linhas. Aí será a entrada do próximo programa, `CASO2.PRG`.

7) Pressione `Ctrl-K` e, a seguir (no menu `BLOCK & SAVE`), tecle a opção `R`, de *insert a file*. Indique o nome do arquivo (`CASO2.PRG`) e você o verá exatamente na posição do cursor, abaixo do arquivo `CASO1.PRG` que trouxemos primeiro. Repita a operação com os outros arquivos até que o último arquivo, `Del.prg`, seja conectado ao grupo de 10 arquivos.

8) Use a opção `E` (`Ctrl-KE`) do menu `BLOCK & SAVE`, para renomear seu novo arquivo, agora um só arquivo contendo todo o grupo de 10 arquivos que interagirão com `central.prg`. Renomeie-o para `AUTOPROC.PRG`, o nome de nosso novo arquivo de procedimentos, que será aberto pelo dBASE juntamente com todos os seus procedimentos, via comando `SET PROCEDURE TO AUTOPROC`, colocado anteriormente no programa de controle, `central.prg`.

Todos os seus programas juntos deverão estar como a documentação que segue. Verifique se você não cometeu nenhum engano.

Caso você prefira, poderá compor seu grupo de *procedures* diretamente no dBASE, porém, cuidado com o tamanho de seu arquivo de *procedures*, pois um arquivo desses só pode ter um número máximo de 4.096 *bytes* (nosso arquivo, a propósito, tem 4.309 caracteres).

```

procedure caso1
set talk off
y = space(1)
clear
@ 10,15 say 'Escolha um numero. Relatorio = 1. Tela = 2: ' get y
read
if y = '2'
  use clientes
  clear
  list
  wait 'Pressione qualquer tecla para retornar ao menu
anterior' to tecla
clear
endif
if y = '1'
  use clientes
  clear
  list to print
  wait 'Pressione qualquer tecla para retornar ao menu
anterior' to tecla
clear
endif
set talk on
return

```

```

procedure caso2
R = space(1)
clear
@ 10,15 say 'Tipo de impressao: Global = 1. Especifica = 2: ' get
R
read
if R = '2'
  do caso2A
endif
if R = '1'
  do caso2B
endif
return

```

```

procedure caso2a
set talk off
clear
use livraria
accept 'Deseja pesquisar em qual campo?: ' to C
accept 'Procurar qual &C: ' to COND
accept 'Deseja imprimir?(S/N): ' to R
clear
if upper(R) = 'S'

```

```
    set print on
endif
go top
do while .not. eof( )
if upper(&C) = upper('&COND')
    ? nome
    ? endereco
    ? trim(cidade) + '-' + estado
    ? cep
    ?
    ?
    ?
endif
skip
enddo
set print off

wait 'Pressione qualquer tecla para retornar ao menu anterior 'to
tecla
clear
set talk on
return
```

```
procedure caso2b
use livraria
set talk off
label form endereca to print
wait 'Pressione qualquer tecla para retornar ao menu anterior 'to
tecla
set talk on
return
```

```
procedure caso3
use vendas
set talk off
y = space(1)
clear
@ 10,15 say 'Escolha um numero. Relatorio = 1. Tela = 2: 'get y
read
if y = '2'
    list
    wait 'Pressione qualquer tecla para retornar ao menu
anterior 'to tecla
clear
endif
if y = '1'
    report form vendas to print
    wait 'Pressione qualquer tecla para retornar ao menu
anterior 'to tecla
clear
endif
set talk on
return
```

```
procedure caso4
```

```

x = 0
do while x < > 4
clear
text
    Atualizar:
    1 = Agenda
    2 = Etiquetas
    3 = Vendas
    4 = Saida
endtext
@ 10,15 say 'Escolha um numero: ' get x picture '9' range 1,4
read
do case
    case x = 1
        do num1
    case x = 2
        do num2
    case x = 3
        do num3
endcase
enddo
return

```

```

procedure num1
use clientes
x = 0
do while x < > 4
clear
text
    Selecione:
    1 = Atualiza (use PgUp PgDn . Ctrl-END para sair)
    2 = Apaga registros (por campos)
    3 = Gera: tela
        relatorio impresso
    4 = Retorna ao Menu Principal
endtext
@ 10,15 say 'Escolha um numero: ' get x picture '9' range 1,4
read
do case
    case x = 1
        append
    case x = 2
        do del
    case x = 3
        do caso1
endcase
enddo
return

```

```

procedure num2
use livraria
x = 0
do while x < > 4
clear
text

```

```
Selecone:
1 = Atualiza (use PgUp PgDn. Ctrl-END para sair)
2 = Apaga registros (por campos)
3 = Gera: tela
        relatorio impresso
4 = Retorna ao Menu Principal
endtext
@ 10,15 say 'Escolha um numero: 'get x picture '9' range 1,4
read
do case
    case x = 1
        append
    case x = 2
        do del
    case x = 3
        do caso2
endcase
enddo
return
```

```
procedure num3
use vendas
x = 0
do while x < > 4
clear
text
    Selecone
    1 = Atualiza (PgUp PgDn. Ctrl-END para sair)
    2 = Apaga registros (por campos)
    3 = Gera: tela
            relatorio impresso
    4 = Retorna ao Menu Principal
endtext
@ 10,15 say 'Escolha um numero: 'get x picture '9' range 1,4
read
do case
    case x = 1
        append
    case x = 2
        do del
    case x = 3
        do caso3
endcase
enddo
return
```

```
procedure del
clear
accept 'Deseja pesquisar em qual campo?: 'to C
accept 'Procurar qual &C: 'to COND
accept 'Deseja ver o registro? (S/N): 'to R
clear
if upper(R) = 'S'
    list for upper(&C) = upper(COND)
```

```

endif
accept 'Quer apaga-lo? (S/N): 'to A
if upper(A) = 'S'
  clear
  delete for upper(&C) = upper(COND)
endif
list
wait '(... ENTER continua) 'to tecla
x = 0
clear
@ 10,15 say 'Confirmar seu apagamento? Sim = 1. Nao = 2.: ' get x
picture '9' range 1,2
read
do case
  case x = 1
    pack
    clear
    list
    wait 'Sua tela atual. (.... ENTER retorna ao Submenu)'
  case x = 2
    recall all
    list
    wait 'Sua tela atual. (.... ENTER retorna ao Submenu)'
return

```

Aproveite para editar seu programa AUTOPROC.PRG se você quiser. A hora é essa!

Grave seu novo arquivo de volta no disquete original (do dBASE) para usá-lo posteriormente. A seguir, recarregue o dBASE novamente para chamar AUTOPROC e trabalhar com central.prg.

Para fazer AUTOPROC funcionar, simplesmente faça como antes, digite:

DO CENTRAL ENTER

e nosso velho programa de menu fará o resto, via comando SET PROCEDURE TO AUTOPROC.

No próximo capítulo abordaremos a criação de telas personalizadas. No Capítulo 15 falaremos sobre Conversões e Interfaces: dBASE II, WordStar, Turbo Basic e Lotus 1-2-3 para dBASE III Plus, além de algumas interações especiais com SideKick e SideWays para realçar ainda mais nossas técnicas de impressão. No Capítulo 16 trataremos de um assunto muito interessante: o Clipper e a compilação de arquivos em dBASE III Plus para arquivos do tipo .EXE, para posterior execução desses mesmos arquivos diretamente no DOS, sem a necessidade do dBASE para orientar sua execução.

Capítulo

14

Criação de Telas Personalizadas

Neste capítulo trabalharemos com o desenvolvimento de telas personalizadas, ou seja, uma maior flexibilidade para usuários ainda pouco familiarizados com operações do dBASE III Plus, principalmente as operações de modificação de arquivos.

Como exemplo, citamos nosso SuperMacro *central.prg*. Lembra-se de que no programa (ou *procedure* – desde o capítulo anterior) de atualização de cada um dos arquivos que fazem parte do sistema, usamos o comando APPEND? Pois bem, a tela de APPEND pode se tornar em uma pequena mas desgastante dor-de-cabeça para o usuário compreender. E nós podemos fazer alguma coisa para evitar isso com o *Screen Painter* do dBASE III Plus.

O SCREEN PAINTER

O *Screen Painter* vem a ser um formatador de telas especial para a exibição de entradas de dados mais fácil de entender, devido a seu formato mais “personalizado”.

Com o *Screen Painter*, pode-se simular o tipo de formulário geralmente utilizado por sua empresa ou atividade específica, inclusive omitindo determinadas entradas que são importantes para o arquivo mas não o são para seu formulário. Tudo isso pode ser informado ao dBASE previamente, de modo que, quando o usuário chamar a tela de APPEND ou de EDIT, essa tela específica apareça no lugar da tela tradicional!

As Formas de Criação

Iniciaremos a criação de nossa primeira tela personalizada agora, via menu. Posteriormente, neste capítulo, explicaremos a maneira simples de se conseguir isso via *prompt*. (E no Capítulo 15, mostraremos com o Sideways, como produzir em um arquivo de comando, a programação completa dessa tela.)

Tecla F2 ou digite ASSIST para entrar no menu principal do dBASE III Plus. A seguir, selecione:

MENU
Create
Format
B:
TELA
ENTER

Seu primeiro menu, ou seja, o Menu Principal do *Screen Painter*, aparecerá como na Figura 14.1.

Agora, vamos selecionar o arquivo fonte de nosso trabalho de personalização de tela.

Neste exercício, iremos reformatar a tela de APPEND do arquivo LIVRARIA.DBF, para

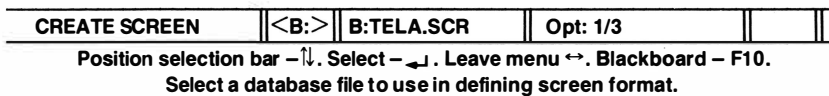
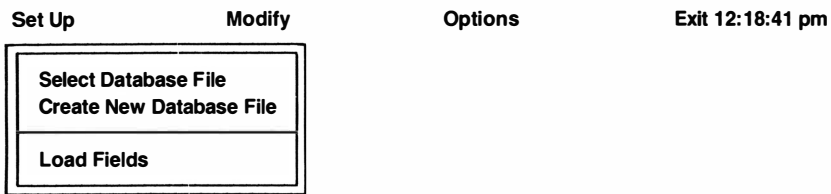


Figura 14.1: Menu Principal do *Screen Painter*.

que, quando este comando for usado (APPEND – e também EDIT), a tela exibida seja uma tela especial, feita exclusivamente para o usuário e não a tela padrão que vem aparecendo cada vez que digitamos APPEND durante nossas interações com este livro (inclusive, como veremos, em nosso programa NUM2.PRG – a parte integrante do SuperMacro central.prg).

Selecione então:

MENU
Set Up
Select Database File
LIVRARIA.DBF
Load Fields
NOME
ENDERECO
CIDADE
ESTADO
CEP
TEL
 ->

e sua tela deverá estar como a tela da Figura 14.2.*

Set Up	Modify	Options	Exit 12:19:57 pm
NOME	XXXXXXXXXXXXXXXXXXXXXXXXXXXX		
ENDERECO	XXXXXXXXXXXXXXXXXXXXXXXXXXXX		
CIDADE	XXXXXXXXXX		
ESTADO	XX		
CEP	XXXXXX		
TEL	XXXXXXXXXX		

CREATE SCREEN	<B:>	B:TELA.SCR	Pg 01 Row 00 Col 00	Ins	
Enter text. Drag field or box under cursor with  . F10 for menu.					
Screen field definition blackboard					

Figura 14.2: *Screen Painter* pronto para edição.

Analizando os Procedimentos

A primeira coisa a observar são as entradas de comandos, como sempre. Na primeira fase, no início, criamos o arquivo TELA.SCR. Este arquivo é um arquivo especial para os trabalhos de formatação, ou seja, de projeto de tela “customizada” que iremos preparar a seguir. Quando aplicada ao trabalho para o qual ela será dedicada, receberá a extensão .FMT (de **ForMaTação**).

Na segunda etapa de nossa série de comandos no menu, vemos a seleção do arquivo-fonte LIVRARIA.DBF, seguido da seleção *Load File* que permite ao usuário optar pelos campos com os quais desejaremos trabalhar na construção de nossa tela personalizada. Os campos escolhidos são, por enquanto, todos os campos exceto o último, REM – que é um campo *memo* de uso especial (já visto anteriormente, mas que será novamente lembrado ainda neste capítulo). A seta para a direita (ou para a esquerda – tanto faz) nos leva então à tela apresentada.

*N.E.: Sua tela poderá estar um pouco diferente, em quantidades de “X”, devido à nossa diagramação/estética.

TRABALHANDO COM OS CAMPOS DE SCREEN PAINTER

A primeira coisa que sugerimos aqui, é acionar a tecla INS, pois usaremos muito sua capacidade de movimentação de caracteres.

Observe também, em nossa segunda tela, que as letras “X” formam o espaço alocado por nós para o preenchimento de cada campo, quando criamos o arquivo LIVRARIA original.

Vamos agora operar com o conteúdo dessa tela.

Agora, com o cursor no canto esquerdo superior da tela, exatamente sob a letra “N” do campo NOME, pressione ENTER 6 vezes. Isto fará com que todo o texto desça um pouco para que possamos passar à próxima operação.

Salvando seu Arquivo Parcial

Salve seu arquivo parcial agora, para que você já possa contar com seu projeto definitivo e prepará-lo para as inserções mais complexas que faremos em seguida.

Para salvar seu arquivo, entre na opção *Exit* do menu do *Screen Painter* (caso você não esteja no menu, pressione F10) e selecione *Save*.

Para rever sua tela especial e adicionar as novas inserções, vá para a opção *Modify* do menu *Assist* e selecione *Format, B:; TELA.FMT* para acessar seu projeto de tela personalizada. Assim que entrar no menu, tecla F10 para obter sua tela de trabalho.

Pressione HOME também 6 vezes até que o cursor chegue à posição Row 00 Col 00, que pode ser controlada via linha de *status*, a parte inferior iluminada da tela (que pode não existir na sua tela, caso você tenha usado o comando SET STATUS OFF anterior à essa operação).

A seguir, pressione a tecla F10 para entrar no menu de formatação especial. Vá para a opção *Options*. Selecione a subopção *Double bar* – que significa *linha contínua dupla* neste contexto. Esta linha servirá para desenhar um retângulo perfeito que abrigará o título de nossa tela. Como diz o texto do menu da Figura 14.3, será uma linha dupla (*double*) dessa vez. (Da Figura 14.3 até a Figura 14.9, não exibiremos a linha de *status*, por razões de estética.)

Após selecionar *Double bar* com ENTER, com o cursor ainda em Row 00 e Col 00, pressione ENTER. Isto irá demarcar o início de nosso retângulo. A seguir, desça com a seta para baixo duas vezes, até a linha dois, ou seja, Row 02. Com a seta à direita, continue (não pressione ENTER ainda) até a coluna 51 (Col 51). Sua posição final deverá ser Pg 01, Row 02 Col 51. Pressione ENTER agora. Note que sua tela exibe um retângulo de linhas duplas como na Figura 14.4. Entre com o cursor no retângulo e escreva Banco de

Dados – DISTRIBUIDORES/BRASIL, ou qualquer outro título que você preferir. Sua tela, ainda parcial, deverá ficar como na Figura 14.5.

Para centralizar o título dentro do quadro, use a barra de espaço com o modo Ins de inserção, com o cursor na primeira letra à esquerda do título. Pressioná-la umas 4 ou 5 vezes dará o resultado esperado.

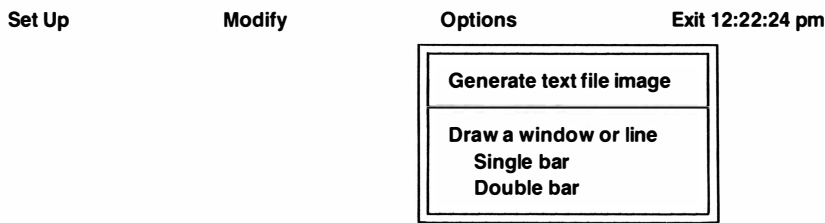


Figura 14.3: Menu de *Options*.

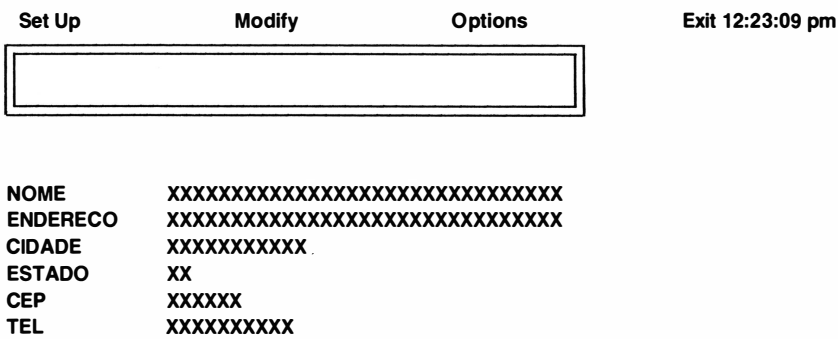


Figura 14.4: Retângulo de Edição com *Screen Painter*.

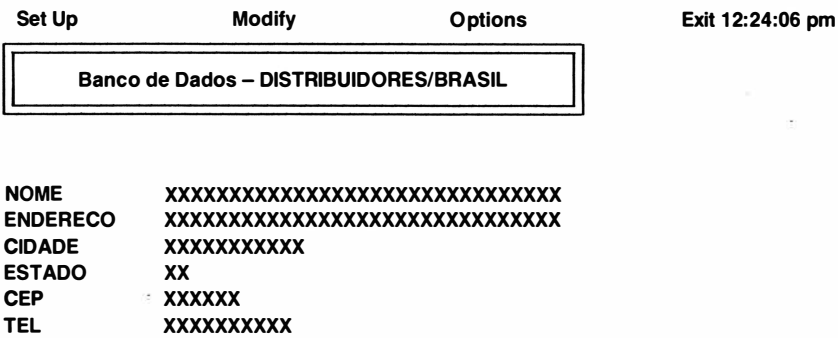


Figura 14.5: Edição Parcial com o *Screen Painter*.

Agora vá para o campo ENDEREÇO e pressione ENTER sob a letra “E” deste campo. Faça o mesmo com o cursor sob a letra “C” de CIDADE para organizar mais os espaços. Faça o mesmo com o “C” de CEP e o “T” de TEL. Consulte sua tela, que deverá estar como na Figura 14.6.

Set Up	Modify	Options	Exit 12:26:26 pm
Banco de Dados – DISTRIBUIDORES/BRASIL			

NOME	XXXXXXXXXXXXXXXXXXXXXXXXXXXX
ENDEREÇO	XXXXXXXXXXXXXXXXXXXXXXXXXXXX
CIDADE	XXXXXXXXXX
ESTADO	XX
CEP	XXXXXX
TEL	XXXXXXXXXX

Figura 14.6: “Personalizando” TELA.SCR.

A seguir, vá para o campo ESTADO com a seta, e, com o cursor sob o campo, tecele END para entrar no modo *drag*, de transporte. Este modo permite que transportemos qualquer campo para o local desejado da tela. Tecle ENTER para confirmar o modo de transporte. Com a seta à direita e depois com a seta para cima, vá para a coordenada Row 10 Col 33, ao lado do campo CIDADE. (Nós iremos inserir o campo ESTADO – SP ao lado de CIDADE, para melhorar sua estética.)

Pressione ENTER nessa coordenada para efetivar o transporte. Note que o espaço alocado para o preenchimento do campo, as duas letras “X” iluminadas, agora está na coordenada Row 10 Col 33, como queremos.

Desça para a linha (Row 12) do campo ESTADO (não é preciso ir até o campo) e tecele Ctrl-Y duas vezes para apagar a linha – e, conseqüentemente, o campo ESTADO – e, ao mesmo tempo, subir a linha abaixo para compensar o espaço deixado por ESTADO.

Talvez você queira saber suas alternativas de controle do cursor. Se quiser, consulte a Tabela 14.1, de Controle do Cursor.

Agora vá para o campo CEP. Pressione END sobre este campo, (ou Home na linha 12, à direita do campo “XXXXXX”) e a seguir ENTER. Vá com as setas até a posição Row 12 Col 42 e quando estiver lá pressione ENTER para confirmar a nova posição de entrada pa-

ra esse campo. Em seguida apague a linha onde se encontra o antigo campo CEP, com Ctrl-Y nessa linha. Acione Ctrl-Y mais uma vez para subir o campo TEL, para efeito de estética.

Sua tela agora deverá estar como na Figura 14.7. Agora vamos preencher nossos dois campos modificados, ESTADO e CEP.

Tecla	Alternativa	Efeito
Setas		Movem o cursor.
Ins	Ctrl-V	Entra/sai do modo de inserção.
Ins		Expande um campo iluminado.
	Ctrl-N	Acrescenta um espaço em branco.
Backspace		Apaga o caractere à esquerda.
Del	Ctrl-G	Apaga o caractere sobre o cursor.
Del		Diminui o tamanho de um campo iluminado.
Home	Ctrl-A	Move o cursor para o início da palavra atual ou anterior.
End	Ctrl-F	Move o cursor para o início da próxima palavra.
ENTER	Ctrl-M	Com Ins ativo, insere uma nova linha.
ENTER		Move o cursor para baixo uma linha.
PgUp	Ctrl-R	Sobe uma página (tela) inteira.
PgDn	Ctrl-C	Desce uma página (tela) inteira.
	Ctrl-T	Apaga palavra à direita.
	Ctrl-Y	Apaga toda a linha.
	Ctrl-U	Apaga todo o registro.
F10		Alterna entre o menu e o formulário.

Tabela 14.1: Teclas de Controle e Edição do *Screen Painter*.

Set Up

Modify

Options

Exit 12:30:08 pm

Banco de Dados – DISTRIBUIDORES/BRASIL

NOME

XXXXXXXXXXXXXXXXXXXXXXXXXXXXX

ENDereco

XXXXXXXXXXXXXXXXXXXXXXXXXXXXX

CIDADE

XXXXXXXXXX

XX

XXXXXX

TEL

XXXXXXXXXX

Figura 14.7: Alterações estéticas.

PREENCHENDO OS NOVOS CAMPOS

Para preencher os novos campos, isto é, escrever na sua tela personalizada, vá para entre os campos que você acabou de transportar no modo *drag* (ESTADO e CEP) com as setas. Sempre no modo *Ins*, escreva os novos campos Estado e depois Cep, na mesma linha onde se encontram seus campos transportados (que estavam sem seus nomes originais). Note que você pode escrever entre os campos com facilidade no modo *Ins*, visto que ele se encarrega de mover os campos iluminados durante a digitação.

Aproveite e reformate os outros campos de sua tela, escrevendo sobre eles (com *Ins* desativado – ou com a combinação de *Ins* e *Del*) os mesmos nomes em letras minúsculas também, até que sua tela fique como na Figura 14.8.

Abaixe mais umas duas linhas, o campo Tel, com ENTER ENTER – no modo *Ins* – sob a letra “T”, por razões de estética.

Set Up	Modify	Options	Exit 12:32:26 pm
Banco de Dados – DISTRIBUIDORES/BRASIL			
Nome	XXXXXXXXXXXXXXXXXXXXXXXXXXXXX		
Endereco	XXXXXXXXXXXXXXXXXXXXXXXXXXXXX		
Cidade	XXXXXXXXXX	Estado XX	Cep XXX
Tel	XXXXXXXXXX		

Figura 14.8: Sua nova tela de TELA.SCR.

Especificando Condições Especiais

Você também pode especificar condições especiais para manipulação de sua tela personalizada. Por exemplo, para dizer ao dBASE que só aceita números no campo Tel, coloque seu cursor no campo Tel, na posição de entrada onde se encontram as letras “X” que marcam o tamanho deste campo (Use End, Home, ou as setas). Com o cursor na posição inicial do campo reservado para os números telefônicos, acesse o menu principal com F10. Observe se o menu está como na Figura 14.9 (se não estiver, verifique se o cursor não foi deixado fora do campo de entrada de Tel).

Pressione ENTER sobre a opção *Picture Template*.

Set Up	Modify	Options	Exit 12:22:24 pm
Screen Field Definition Action : Edit/SET Source : LIVRARIA Content : TEL Type :Character Width : 14 Decimal: Picture Function: Picture Template: Range:			

Figura 14.9: Menu *Modify*, *Content*, *TEL*, *Picture Template*.

A seguir, nesta opção, digite:

(999)999-9999 ENTER

que servirá de entrada para nosso objetivo de somente deixar que números nesse formato de apresentação entrem na resposta do usuário. Note também que este comando (*Picture Template*) funciona mais ou menos como o comando **PICTURE** usado no capítulo anterior, nos programas do SuperMacro *central.prg* e em *autoproc.prg*. Observe também a tabela de consulta à direita da tela do menu *Modify*, como na Figura 14.10, que traz uma explicação dos códigos que podem ser usados nessa modalidade de opção. Para reforçar seu domínio, consulte sua tradução na Tabela 14.2.

A	Quaisquer caracteres alfabéticos.
L	Somente T (<i>True</i>), F (<i>False</i>), Y (<i>Yes</i>) e N (<i>No</i>).
N	Dígitos e caracteres alfabéticos.
X	Qualquer caractere.
Y	Somente Y ou N.
#	Dígitos, espaços, símbolos e pontos.
9	Dígitos e símbolos.
!	Converte para maiúsculas.
outros	Sobregrava dados a não ser que a função @R seja usada.

Tabela 14.2: Atributos de *Picture Template*.

Agora vamos usar a segunda modalidade de *Picture*. Vá para o campo “XX” de ESTADO, com F10 e suas teclas de controle do cursor. Com o cursor no início do campo “XX”, vá para a opção *Modify* novamente com F10. Passe para *Picture Function* e pressione ENTER sobre esta opção. Sua tela deverá estar como na Figura 14.11.

Set Up

Modify

Options

Exit 12:35:12 pm

Screen Field Definition

Action : Edit/GET

Source : LIVRARIA

Content : TEL

Type :Character

Width : 14

Decimal:

Picture Function:

Picture Template:

Range:

Character Input Symbols

A Any alpha character

L Allow, T, F, Y or N

N Alpha and digits

X Any character

Y Allow Y, or N

Allow digits, spaces, signs, and periods

9 Allow digits and signs

! Convert to uppercase

other Overwrite data unless @R function is used

Picture value (999)999-9999

MODIFY SCREEN

<B:>

B:TELA.SCR

Pg 01 Row 14 Col 12

Ins

Enter a picture template without using quotes. Finish with .

Enter a picture template for editing or displaying this field.

Figura 14.10: Tela de *Picture Template*.

Insira “!” (sem aspas) para indicar que você deseja converter sua entrada em letras maiúsculas, ou seja, qualquer entrada será aceita, pois todas serão convertidas! Pressione ENTER e você já estará pronto para usar sua tela personalizada. Se for preciso, como no exemplo anterior, consulte a Tabela 14.3 de *Picture Function*.

!	Converte para maiúsculas.
A	Exibe somente caracteres alfabéticos.
D	Data do tipo <i>American</i> : mm/dd/yy.
E	Data do tipo <i>European</i> : dd/mm/yy.
S	Movimento horizontal da tela.
R	Insere caractere, não sobregrava.

Tabela 14.3: Atributos de *Picture Function*.

Salvando seu Trabalho (novamente)

Para salvar todo o seu trabalho, vá para a opção *Exit* do menu principal (caso você não

Set Up	Modify	Options	Exit 12:37:43 pm
--------	--------	---------	------------------

Screen Field Definition

Action : Edit/GET

Source : LIVRARIA

Content : ESTADO

Type :Character

Width : 2

Decimal:

Picture Function:

Picture Template:

Range:

Character Input Functions

! convert to uppercase

A display only alpha chars

D American mm/dd/yy date

E European dd/mm/yy date

S horizontal scrolling

R insert <other> char
don't overwrite it

Function value!

MODIFY SCREEN	<B:>	B:TELA.SCR	Pg 01 Row 10 Col 37 Ins
---------------	------	------------	-------------------------

Enter one or more function symbols without using quotes. Finish with ↵.

Enter a picture function for editing or displaying this field.

Figura 14.11: Tela de *Picture Function*.

esteja no menu agora, simplesmente tecle F10) e pressione ENTER sobre a subopção *Save*. Todo o seu trabalho estará a salvo e armazenado no disquete. Não se esqueça de fazer uma cópia de seu disquete para não correr o risco de perder todo esse trabalho.

Modificando sua Tela Personalizada

Para modificar seu arquivo TELA.SCR (que em uso receberá a extensão de arquivo .FMT), vá para a opção **Modify** do menu *Assist* e selecione:

```

MENU
Modify
Format
B:
TELA.SCR
F10

```

e você poderá trabalhar normalmente em sua tela para mais modificações. O processo de armazenamento será o mesmo sempre, ou seja: **Exit** e **Save**.

Operando com sua Tela Personalizada

Para operar com sua tela personalizada, você deve acessar o menu principal – *Assist* – e selecionar:

MENU
Set Up
Format for Screen
B:
TELA.FMT

Agora, com *Edit* ou *Append*, esta será sua tela de trabalho. Se você quiser usá-la, simplesmente acione *Append* ou *Edit* e você terá sua tela personalizada imediatamente. Aproveite e vá para o Registro 15 com PgUp, como prática. (Para sair – e salvar – do modo *Append* ou *Edit*, tecle Ctrl-End.) Veja nas Figuras 14.12 e 14.13, as duas telas de *Edit*/*Append*.

Banco de Dados – DISTRIBUIDORES/BRASIL
--

Nome

Endereco

Cidade

Estado

Cep

Tel () -

APPEND	<B:>	LIVRARIA	Rec: 18/19	Ins
--------	------	----------	------------	-----

Figura 14.12: Tela personalizada de *Append/Edit* vazia.

Alterando o Tamanho dos Campos

Alteração Definitiva

Você também pode alterar o tamanho de seus campos, de forma definitiva ou permanente. Para isso, basta entrar na opção **Modify** do menu *Assist* (se você se encontrar fora do *Screen Painter*) e proceder como mostramos há pouco, para reentrar em TELA.SCR. Vá para a opção **Modify** do menu do *Screen Painter* (caso seu menu não funcione, pressione *Width*). Porém, há um pequeno detalhe: é necessário “usar” um campo, ou seja, definí-lo primeiro, para poder fazer *Width* funcionar. (Cuidado, esta opção muda o tamanho do campo *permanentemente*!)

Banco de Dados – DISTRIBUIDORES/BRASIL
--

Nome EBRAS – Editora Brasileira Ltda

Endereco R. Borges Lagoa, 1044

Cidade Sao Paulo Estado SP Cep 04038

Tel (011)549-8633

APPEND	<B:>	LIVRARIA	Rec: 15/19	Ins
--------	------	----------	------------	-----

Figura 14.13: Tela personalizada de *Append/Edit* no Registro 15.

Você também pode proceder da mesma forma para alterar possíveis campos que tenham números decimais programados, para mais ou menos casas decimais. Mas lembre-se, essas mudanças (via *Width*) serão feitas na estrutura do arquivo também.

Alteração Provisória

É possível aumentar ou diminuir o tamanho de um campo também provisoriamente (isto é, sem modificar a estrutura permanente do arquivo), somente do arquivo especial TELA.SCR.

Para modificar os campos provisoriamente, entre na região iluminada onde se encontram as letras “X”, e pressione Ins. A cada pressionamento de Ins o tamanho do campo aumentará de um caractere. Ctrl-End salvará suas alterações provisórias da tela.

Acrescentando mais Campos em sua Tela Personalizada

Pode-se também acrescentar (ou eliminar) mais campos (contanto que seja do arquivo-fonte) no arquivo TELA.SCR, de modo que, quando você for utilizá-lo sob o nome TELA.FMT, essas modificações estejam na sua tela. Para isso, na tela de edição de TELA.SCR (a que tem os campos “XXXX...” e com o cursor na posição de entrada do novo campo, pressione F10 para entrar no menu do *Screen Painter*. (Lembre-se, caso o menu *Modify* não funcione, pressione F10 duas vezes para ativá-lo.) Selecione:

MENU
Modify

Content
REM
F10

e você verá seu novo campo (REM) na posição marcada anteriormente com o cursor, como na Figura 14.14.

Set Up	Modify	Options	Exit 12:46:28 pm					
Banco de Dados – DISTRIBUIDORES/BRASIL								
Nome	XXXXXXXXXXXXXXXXXXXXXXXXXXXX							
Endereco	XXXXXXXXXXXXXXXXXXXXXXXXXXXX							
Cidade	XXXXXXXXXX	Estado XX	Cep XXX					
Tel	(999)999-9999							
MEMO								
<table border="1"> <tr> <td>MODIFY SCREEN</td> <td><B:></td> <td>B:TELA.SCR</td> <td>Pg 01 Row 16 Col 00</td> <td>Ins</td> </tr> </table>				MODIFY SCREEN	<B:>	B:TELA.SCR	Pg 01 Row 16 Col 00	Ins
MODIFY SCREEN	<B:>	B:TELA.SCR	Pg 01 Row 16 Col 00	Ins				
Enter text. Drag field or box under cursor with ↵ . F10 for menu.								
Field: LIVRARIA -> REM Type: Memo Width: 10								

Figura 14.14: Screen Painter com MEMO.

Apagando Campos

Permanentemente

Para apagar um campo permanentemente, inclusive da estrutura do arquivo (LIVRARIA.DBF), use Ctrl-U sobre o campo desejado. Uma mensagem do dBASE perguntará se você realmente quer fazer isto. “Y” apagará seu registro. “N” apagará somente de sua tela. Não use este comando quando não quiser correr o risco de apagamentos indevidos.

Provisoriamente

Ctrl-Y apagará seu registro provisoriamente, ou seja, somente na tela. Sua estrutura geral ainda conterà o campo apagado com esta combinação de teclas.

Usando o campo MEMO com Screen Painter

Já vimos como acessar um campo memo de um registro. Agora, achamos oportuno revermos essa operação novamente, com o *Screen Painter*.

Para acessar o campo memo de nosso primeiro registro, vá para o menu *Assist*, na posição *Update*. Selecione *Edit*. Mova-se para o Registro 15 (caso você tenha indexado seu arquivo LIVRARIA.DBF, o Registro é o da EBRAS – Editora Brasileira). Sua tela deverá estar como na Figura 14.15.

Banco de Dados – DISTRIBUIDORES/BRASIL
--

Nome EBRAS – Editora Brasileira Ltda

Endereco R. Borges Lagoa, 1044

Cidade Sao Paulo Estado SP Cep 04038

Tel (011)549-8633

MEMO

EDIT	<B:>	LIVRARIA	Rec: 15/26	Ins
------	------	----------	------------	-----

Figura 14.15: Nosso primeiro registro de TELA.FMT com Edit.

A seguir, coloque seu cursor no campo memo (REM) e tecle Ctrl-PgDn.

Sua tela deverá mostrar uma área de trabalho parecida com nossa tela de MODIFY COMMAND, escrito Edit: REM na parte esquerda superior do vídeo, como na Figura 14.16. Aproveite para corrigir a palavra “informática” que deverá estar em nosso Registro da Ebras, feito anteriormente (ou copie toda a frase da Figura 14.16), colocando a primeira letra em maiúscula. Em seguida, armazene tudo com Ctrl-W (como em MODIFY COMMAND, lembra-se?). Se quiser ver o conteúdo de seu campo memo específico do registro desta Editora, tecle Ctrl-PgDn novamente. Tecle Ctrl-W e salve tudo de novo.

Criando uma Tela Especial de Auxílio

Para “descomplicar” um pouco a vida do usuário, você pode projetar uma tela especial de auxílio permanente, para guiá-lo durante suas interações com seu programa. E é isto o que faremos a partir de agora.

Edit: REM		Ins	
CURSOR <-- --> Char: ← → Word: Home End Line: ^← ^→ Reformat: ^KB		UP DOWN Line: ↑ ↓ Page: PgUp PgDn Find: ^KF Refind: ^KL	
DELETE Char: Del Word: ^T Line: ^Y		Insert Mode: Ins Insert Line: ^N Save: ^W Abort: Esc Read file: ^KR Write file: ^KW	

Edicao de livros de informatica de alta qualidade.

Figura 14.16: Acesso direto ao campo MEMO com Ctrl-PgDn.

Entre no menu **Modify** de *Assist* e selecione as opções *Format*, *B:* e *TELA.SCR*. Depois, com F10, acesse sua tela de trabalho. Em seguida, coloque o cursor na posição Row 05 Col 54, a posição onde iniciaremos nosso desenho do retângulo onde permanecerá a tela de auxílio. Com o cursor nesta posição, tecle F10. A seguir, selecione:

MENU
Options
Double bar

Após selecionar *Double bar*, que significa que escolhemos uma linha contínua dupla para nosso desenho – e a tela com nossos registros do tipo “XXXXXX...” tiver retornado –, pressione ENTER novamente para marcar a posição de início do desenho. Depois, com a seta para baixo, vá até a coordenada Row 19 (não pressione ENTER ainda!). A seguir, com a seta para a direita, vá até a coordenada Col 79 e finalmente pressione ENTER. Sua tela deverá estar como nossa Figura 14.17. Entre com o cursor no retângulo e, iniciando na posição Row 06, escreva as informações contidas na Figura 14.18. Cuidado! Não pressione ENTER no modo Ins ou o conteúdo de sua tela inteira se moverá do lugar, abaixando uma linha.

Para fazer os traços horizontais use, para cada traço, a combinação F10 – *Options* – *Single bar* e marque o início de cada linha.

Set Up	Modify	Options	Exit 12:49:31 pm
Banco de Dados – DISTRIBUIDORES/BRASIL			
Nome	XXXXXXXXXXXXXXXXXXXXXXXXXXXX		
Endereco	XXXXXXXXXXXXXXXXXXXXXXXXXXXX		
Cidade	XXXXXXXXXX	Estado	XX Cep XXX
Tel	(999)999-9999		
MEMO			
MODIFY SCREEN<B:>B:TELA.SCRPg 01 Row 19 Col 79Ins Enter text. Drag field or box under cursor with . F10 for menu. Screen field definition blackboard			

Figura 14.17: Tela de auxílio inicial.

Um detalhe importante: a tela que acabamos de construir só servirá para uso com os comandos *Append* e *Edit*, do menu ou do *prompt*.

Lembre-se também: para operar normalmente com sua tela personalizada, primeiro você deve inicializá-la com:

```
MENU
Set Up
Format for Screen
B:
TELA.FMT
```

Experimente trabalhar com sua tela completa. Salve tudo com *Exit* e *Save* e entre na opção *Update* do menu *Assist* para, a seguir, selecionar a opção *Edit* e visualizar sua nova tela personalizada.

Mais Opções do Menu Screen Painter

Já vimos quase tudo a respeito de telas personalizadas. Agora vamos terminar de analisar seu menu para depois tratarmos das entradas (facílimas) a partir do *prompt* e, finalmente,

Set Up	Modify	Options	Exit 12:59:39 pm					
Banco de Dados – DISTRIBUIDORES/BRASIL								
Nome XXXXXXXXXXXXXXXXXXXXXXXXXXXX Endereco XXXXXXXXXXXXXXXXXXXXXXXXXXXX Cidade XXXXXXXXXX Estado XX Cep XXX Tel (999)999-9999 MEMO	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center; padding: 5px;">Auxílio na Tela</td> </tr> <tr> <td style="padding: 5px;"> Cursor: Setas Paginas: PgDn e PgUp </td> </tr> <tr> <td style="padding: 5px;"> Apagar: Letra = Del Campo = Ctrl-Y Registro = Ctrl-U </td> </tr> <tr> <td style="padding: 5px;"> Memo Ctrl-PgDn (dentro) </td> </tr> <tr> <td style="padding: 5px;"> Salvar: Ctrl-End Desistir: Esc </td> </tr> </table>			Auxílio na Tela	Cursor: Setas Paginas: PgDn e PgUp	Apagar: Letra = Del Campo = Ctrl-Y Registro = Ctrl-U	Memo Ctrl-PgDn (dentro)	Salvar: Ctrl-End Desistir: Esc
Auxílio na Tela								
Cursor: Setas Paginas: PgDn e PgUp								
Apagar: Letra = Del Campo = Ctrl-Y Registro = Ctrl-U								
Memo Ctrl-PgDn (dentro)								
Salvar: Ctrl-End Desistir: Esc								
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">MODIFY SCREEN</td> <td style="border: 1px solid black; padding: 2px;"><B:></td> <td style="border: 1px solid black; padding: 2px;">B:TELA.SCR</td> <td style="border: 1px solid black; padding: 2px;">Pg 01 Row 18 Col 73</td> <td style="border: 1px solid black; padding: 2px;">Ins</td> </tr> </table> Enter text. Drag field or box under cursor with . F10 for menu. Screen field definition blackboard				MODIFY SCREEN	<B:>	B:TELA.SCR	Pg 01 Row 18 Col 73	Ins
MODIFY SCREEN	<B:>	B:TELA.SCR	Pg 01 Row 18 Col 73	Ins				

Figura 14.18: Tela final personalizada.

fazermos uma pequena modificação num dos programas do SuperMacro `central.prg`, inserindo nele dois comandos para que ele também use nossa nova tela formatada.

O menu **Modify**, do *Screen Painter* (não confundir com a opção de mesmo nome do *Assist*), tem mais uma opção com a qual não trabalhamos neste capítulo, *Range*.

Range funciona mais ou menos como seu homônimo usado na programação do SuperMacro `central.prg`. Ele serve para delimitar limites máximos e mínimos para uma determinada entrada. Sua forma de operação de inserção é idêntica. Faça sua própria experiência com *Range*, de preferência com um arquivo que tenha limites numéricos máximos e mínimos, como o arquivo `VENDAS.DBF`, como no exemplo da Figura 14.19. Veja também na Tabela 14.4 o conteúdo da tabela de entrada desta opção.

Input Range	Faixa de entrada.
Lower Limit:	Limite inferior.
Upper Limit:	Limite superior.

Tabela 14.4: Tabela de Inserção de *Range*.

Set Up	Modify	Options	Exit 11:45:21 pm					
Screen Field Definition Action : Edit/GET Source : VENDAS Content : QTDE Type : Numeric Width : 3 Decimal: 0 Picture Function: Picture Template: Range: Input Range Lower Limit Upper Limit:								
<table border="1" style="width: 100%; border-collapse: collapse;"><tr><td style="width: 25%;">CREATE SCREEN</td><td style="width: 25%; text-align: center;"><B:></td><td style="width: 25%;">B:VENDAS.SCR</td><td style="width: 25%; text-align: center;">Opt: 1/2</td><td style="width: 25%; text-align: center;">Ins</td></tr></table> Enter new value. Finish with ↵. Enter the maximum and minimum values allowed for this field.				CREATE SCREEN	<B:>	B:VENDAS.SCR	Opt: 1/2	Ins
CREATE SCREEN	<B:>	B:VENDAS.SCR	Opt: 1/2	Ins				

Figura 14.19: Exemplo de interação com o menu *Range*.

Agora, para finalizar, vá para o menu *Options* e pressione ENTER sobre a subopção *Generate text file image*, a primeira deste menu. Se por acaso você pensa que nada aconteceu está muito enganado. Você acabou de criar um arquivo especial de texto, do tipo .TXT, para ser lido por seu editor de texto, contendo a definição dos campos e posições de sua tela personalizada (no Capítulo 15 falaremos de Conversões e Interfaces e aproveitaremos para “ler” este arquivo que será a forma programada de TELA.SCR).

Ainda antes de finalizar, vamos analisar algumas subopções do menu *Modify* que não foram usadas ainda.

Action significa que, ao pressionarmos ENTER, ativamos/desativamos os modos *Edit/GET* e *Display/SAY*. No primeiro, podemos modificar os campos via comandos *Edit* ou *Append*, já vistos antes neste capítulo. No modo *Display/SAY*, as informações somente podem ser lidas e não podem ser modificadas.

Source significa o arquivo-fonte de toda a operação. Neste capítulo, por exemplo, operamos com nosso velho conhecido LIVRARIA.DBF – a *fonte* de nossos dados.

Type, a última opção ainda não mencionada, significa o tipo dos dados a serem trabalhados, tal qual na estruturação de um arquivo: Caracteres, Numéricos, Data, Lógico e Memo.

TRABALHANDO COM O SCREEN PAINTER A PARTIR DO PROMPT

Podemos executar rapidamente as mesmas operações de criação e manipulação de uma te-

la personalizada a partir do *prompt*. Por exemplo, para criarmos o mesmo arquivo TELA.SCR que construímos neste capítulo, teríamos de digitar:

MODIFY SCREEN TELA ENTER

e nosso arquivo – novo é claro – teria sido criado. O restante dos procedimentos operacionais seriam os mesmos, uma vez que os arquivos (TELA.SCR e TELA.FMT) já estariam criados e os menus de operação seriam exatamente os mesmos.

Para inicializar sua tela personalizada depois de pronta (após ter sido trabalhada de acordo com as instruções do *Screen Painter* que acabamos de executar), para poder usá-la efetivamente, você teria de digitar:

SET FORMAT TO TELA ENTER

e, em seguida, seu comando de trabalho: EDIT ou APPEND para visualizar seu resultado (não se esqueça de SET DEFAULT TO B, caso seu disco de trabalho esteja no acionador B e você ainda não tenha acionado este comando).

Após trabalhar com sua tela personalizada, você teria de sair do modo APPEND ou EDIT do mesmo modo que o fizemos anteriormente, ou seja, com Ctrl-End, de acordo com o menu de auxílio que você teria criado também.

Apenas um pequeno detalhe: após sair de sua tela personalizada, você teria de acionar o comando:

CLOSE FORMAT ENTER

para fechar sua tela do *Screen Painter*. Para melhor ilustrar isso – e para manter o padrão de didática que viemos mantendo até agora –, vamos aplicar esses dois comandos (abertura e fechamento) em um nosso conhecido colaborador: o programa NUM2.PRG do complexo de programas central.prg (ou autoproc.prg, tanto faz!).

Primeiro não se esqueça de salvar seu trabalho adequadamente. Depois, acesse o programa NUM2.PRG com:

MODIFY COMMAND NUM2 ENTER

e modifique o pequeno trecho abaixo de *case x = 1* até que ele fique assim (mantenha o restante do programa intacto!).

```
case x = 1
    set format to tela
    append
    close format
```

Compare seu novo programa NUM2.PRG com a Figura 14.20.

```

***** Programa NUM2.PRG - Chamado de CASO4.PRG
***** Atualiza arquivo LIVRARIA.DBF

use livraria
x = 0
do while x < > 4
clear
text
  Seleccione:
  1 = Atualiza (use PgUp PgDn. Ctrl-END para sair)
  2 = Apaga registros (por campos)
  3 = Gera: tela
      relatorio impresso
  4 = Retorna ao Menu Principal
endtext
@ 10,15 say 'Escolha um numero: ' get x picture '9' range 1,4
read
do case
  case x = 1
    set format to tela
    append
    close format
  case x = 2
    do del@
  case x = 3
    do caso2
endcase
enddo
return

```



Figura 14.20: Novo NUM2.PRG com TELA.FMT.

Agora, quando você for usar seu SuperMacro central.prg (ou autoproc.prg – se você o alterar também), acione a instrução:

DO CENTRAL ENTER

e escolha a opção 4 do menu central, *Atualização*. Como sempre, seu programa o transferirá para o subprograma CASO4.PRG. Escolha, neste próximo menu, a opção 2, *Etiquetas*. A seguir, você será transferido para NUM2.PRG. Agora, em seu terceiro menu, escolha a opção 1, *Atualiza*, que se encarregará de fazer o resto (e melhorar muito nossa apresentação).

Mais um detalhe: provavelmente o retângulo que contém o título “Banco de Dados – DISTRIBUIDORES/BRASIL” apareceu com uma parte “comida”. Isto aconteceu porque retiramos a linha de *status* no programa central.prg e as informações ali contidas passaram a ocupar o mesmo espaço de parte de seu retângulo/título. Você pode resolver isso rapida-

mente trabalhando na programação de `central.prg` com `MODIFY COMMAND CENTRAL`, inserindo a linha `SET STATUS ON` no lugar de `OFF` e retirando a linha `SET STATUS ON` no final do programa, já que não precisará mais dela novamente.

Como dissemos antes, no próximo capítulo falaremos de conversões e interfaces entre os principais programas do mercado internacional e o `dBASE III Plus`. No Capítulo 16, trataremos de outro assunto “quente”: `CLIPPER` – Compilando seus Sistemas. No Capítulo 17, incluiremos o tópico `LAN` – Redes Locais com o `dBASE III Plus` (o assunto do momento!). Até lá (e não se esqueça de fazer uma cópia de seu disquete de trabalho).

Capítulo

15

Conversões e Interfaces

Neste capítulo demonstraremos, com exemplos práticos, como intercambiar arquivos entre o dBASE III Plus e alguns dos mais populares *softwares* disponíveis no mercado internacional. Usaremos, para nossas transferências, os arquivos LIVRARIA.DBF, TELA.SCR (na verdade TELA.TXT, criado na opção *Options*, no capítulo anterior e TELA.FMT, também criado neste capítulo), central.prg, VENDAS.DBF – todos fartamente comentados neste livro – e seus equivalentes criados especialmente para os programas receptores.

Os resultados impressos neste capítulo serão, em alguns casos, mostrados na sua concepção original, isto é, mostraremos uma reprodução direta da impressora, ao contrário do que fizemos por todo o livro, por motivo de estética, com exceção do capítulo referente ao DOS, no início.

Antes de terminar o capítulo, mostraremos como fazer essas transferências ainda no próprio dBASE, via comandos do *prompt* bem como transferências dos programas aqui exemplificados para o dBASE.

Com o SIDEWAYS, da Funk Software, Inc., trabalharemos exclusivamente com suas características de impressão de formatações específicas e iremos imprimir a documentação do arquivo TELA.FMT, o arquivo TELA.TXT e central.prg com caracteres expandidos.

Com o SideKick, da BORLAND Inc., de acordo com o que também prometemos anteriormente, vamos listar (LIST ou *List*) nosso arquivo VENDAS.DBF e, no meio da listagem do programa na tela, chamaremos seu menu e o instruiremos a editar este arquivo para depois imprimi-lo com algumas mudanças – tudo isso sem que você precise sair do dBASE!

Com o excelente Turbo Basic, da Borland International, Inc., vamos utilizar suas características de edição de texto e vamos fazer uma transferência de arquivos com central.prg (arquivos do tipo .TXT também são muito bem-vindos!).

Com a superplanilha Lotus 1-2-3 Versão 2, da Lotus Development Corporation, transferiremos o arquivo VENDAS.DBF e também criaremos um pequeno arquivo no Lotus para depois transferi-lo para o arquivo CLIENTES.DBF no dBASE.

Finalmente, com o não menos excelente WordStar Professional Release 4, da MicroPro International Corporation, transferiremos o arquivo LIVRARIA.DBF e criaremos: 1) uma carta personalizada para ser impressa pelo WordStar para todos os registros contidos no arquivo; 2) endereçamento automático para seus envelopes; 3) um sistema automático de endereçamento de etiquetas. Desta forma, quando você terminar, terá feito quatro sistemas de endereçamento (dois no dBASE com os programas .PRG e dois com a conversão dBASE-WordStar) e um sistema de geração de cartas personalizadas – tudo via arquivo LIVRARIA.DBF.

Antes de terminarmos, ainda falaremos da conversão dBASE II-dBASE III Plus, para fecharmos o capítulo “com chave de ouro”.

Note que não comentaremos os comandos dos programas usados nas conversões e sim os

comandos do dBASE usados para efetuar essas conversões. Entretanto, ilustraremos todos os procedimentos necessários para que você possa trabalhar com tranquilidade – e eficiência –, visto que todos os programas foram testados várias vezes nesta edição brasileira. Portanto, para ser bem sucedido nas operações que apresentaremos a seguir, bastará ter o programa para conversão e prestar atenção no texto copiando corretamente suas instruções.

INTERFACE dBASE III Plus – SIDEWAYS

Com o SIDEWAYS você pode imprimir rapidamente um programa do dBASE III Plus, com características especiais de impressão. Como exemplo, geraremos a documentação do programa TELA.FMT que será gerado com sua formatação de comandos @ SAY, GET e TO.

Para nosso primeiro exemplo com o SIDEWAYS (haverá mais dois), carregue o SIDEWAYS no acionador A, digitando:

SIDEWAYS		ENTER
S I D E W A Y S Version 2.01		Ser#3475318
(C) Funk Software, Inc. 1982		E100
Vertical form size (inches):	8.00	Printer select: LPT1:
Horizontal form size (inches):	8.00	
Double strike:	ON	
Character font:	VERY SMALL	4 x 12 dot matrix
Character spacing (dots):	1	14.40 characters per inch
Line spacing (dots):	3	8.00 lines per inch
Left margin (inches):	2.00	
Max printing width (inches):	10.00	144 characters per line
Top margin (inches):	1.00	
Printing length (inches):	6.00	48 lines per page
Enter name of print file:	B:TELA.FMT	Starting page: 1
		Glue lines: 0
Strike "" and "" to move from field to field		
"F1" to save current options		
"F10" to exit from program		

Figura 15.1: Menu do SIDEWAYS com TELA.FMT.

Sua tela do SIDEWAYS aparecerá com as informações padrão. Com o disquete do dBASE no acionador B, copie as instruções da tela do SIDEWAYS de acordo com a Figura 15.1. (Para alterar a opção da linha *Character font*: (fonte de impressão), tecle F2. Você verá as opções alternarem-se até a opção desejada (as setas o levarão por todo o menu utilizável.):

Após copiar os parâmetros apresentados no menu da Figura 15.1, prepare sua impressora e tecle ENTER.

Sua impressão de TELA.FMT deverá estar como na Figura 15.2.

Para obter toda a estrutura do arquivo TELA.SCR, criado no capítulo anterior – e já transformado em TELA.TXT durante nossas interações com o *Screen Painter*, copie as instruções da Figura 15.3.

Após copiar os parâmetros apresentados no menu da Figura 15.3, prepare sua impressora e tecle ENTER.

Sua impressão de TELA.TXT deverá estar como na Figura 15.4.

```

@ 1, 6 SAY "Banco de Dados - DISTRIBUIDORES/BRASIL"
@ 6, 0 SAY "Nome"
@ 6, 12 GET LIVRARIA->NOME
@ 6, 59 SAY "Auxilio na Tela"
@ 8, 0 SAY "Endereco"
@ 8, 12 GET LIVRARIA->ENDERECO
@ 8, 56 SAY "Cursor: Setas"
@ 9, 56 SAY "Paginas: PgDn e PgUp"
@ 10, 0 SAY "Cidade"
@ 10, 12 GET LIVRARIA->CIDADE
@ 10, 29 SAY "Estado"
@ 10, 37 GET LIVRARIA->ESTADO FUNCTION "!"
@ 10, 41 SAY "Cep"
@ 10, 46 GET LIVRARIA->CEP
@ 11, 56 SAY "Apagar: Letra = Del"
@ 12, 65 SAY "Campo = Ctrl-Y"
@ 13, 62 SAY "Registro = Ctrl-U"
@ 14, 0 SAY "Tel"
@ 14, 12 GET LIVRARIA->TEL PICTURE "(999)999-9999"
@ 15, 56 SAY "Memo: Ctrl-PgDn(dentro)"
@ 16, 0 GET LIVRARIA->REM
@ 17, 56 SAY "Salvar: Ctrl-End"
@ 18, 56 SAY "Desistir: Esc"
@ 0, 0 TO 2, 51 DOUBLE
@ 5, 54 TO 19, 79 DOUBLE
@ 7, 55 TO 7, 78
@ 10, 55 TO 10, 78
@ 14, 55 TO 14, 78
@ 16, 55 TO 16, 78

```

Figura 15.2: Resultado da impressão de TELA.FMT via SIDEWAYS.

SIDEWAYS Version 2.01
(C) Funk Software, Inc. 1982

Ser#3475318
E100

```
Vertical form size (inches):  8.00      Printer select:  LPT1:
Horizontal form size (inches): 8.00

Double strike:                OFF
Character font:               VERY SMALL      4 x 12 dot matrix

Character spacing (dots):      1            14.40 characters per inch
Line spacing (dots):          3            8.00 lines per inch

Left margin (inches):         2.00
Max printing width (inches):  10.00        144 characters per line

Top margin (inches):          1.00
Printing length (inches):     6.00        48 lines per page

Enter name of print file:  B:TELA.TXT      Starting page:    1
                                           Blue lines:      0

Strike "" and "" to move from field to field
"F1" to save current options
"F10" to exit from program
```

Figura 15.3: Menu do SIDEWAYS com TELA.TXT.

Agora, toda a estrutura de TELA.TXT está armazenada no SIDEWAYS à disposição de eventuais alterações na forma de apresentação. O autor utiliza muito seus recursos para impressões “decorativas”, além de rápidas, de seus programas em dBASE (mas somente os programas do tipo .TXT, .PRG e .FMT). Vejamos então um exemplo bem “decorativo”.

Copie, mais uma vez, a tela da Figura 15.5. Atenção ao fornecer os parâmetros de *Character font* (LARGE) e do nome do arquivo a ser lido do acionador B, desta vez, central.prg.

Pressione ENTER e seu arquivo deverá ser impresso como no exemplo da Figura 15.6.

O resultado da programação feita de acordo com a Figura 15.5 é uma impressão estilizada de nosso velho conhecido central.prg! Use o SIDEWAYS para os tipos de impressão que mais se adaptarem a suas necessidades. Passemos agora ao SideKick.

Field definitions for Screen : B:TELA.scr

Page	Row	Col	Data Base	Field	Type	Width	Dec
1	6	12	LIVRARIA	NOME	Character	40	
1	8	12	LIVRARIA	ENDEREÇO	Character	40	
1	10	12	LIVRARIA	CIDADE	Character	15	
1	10	37	LIVRARIA	ESTADO	Character	2	
FUNCTION !							
1	10	46	LIVRARIA	CEP	Character	6	
1	14	12	LIVRARIA	TEL	Character	14	
PICTURE (999)999-9999							
1	16	0	LIVRARIA	REM	Memo	10	

Content of page : 1

Banco de Dados - DISTRIBUIDORES/BRASIL

Nome	XX	Auxilio na Tela
Endereco	XX	Cursor: Setas
Cidade	XXXXXXXXXXXXXXXXXXXX Estado XX Cep XXXXXX	Paginas: PgDn e PgUp
		Apagar: Letra = Del
		Campo = Ctrl-Y
		Registro = Ctrl-U
Tel	XXXXXXXXXXXXXXXXXXXX	
XXXX		Memo: Ctrl-PgDn(dentro)
		Salvar: Ctrl-End
		Desistir: Esc

Figura 15.4: Resultado da impressão de TELA.TXT via SIDEWAYS.

INTERFACE dBASE III Plus – SideKick

Aqui usaremos o SideKick, da BORLAND, Inc., para gerar telas de arquivos do dBASE III Plus, diretamente do programa, ou seja, não precisaremos de sair do dBASE para acessar o SideKick e editar/imprimir um arquivo gerado na tela do dBASE via comando LIST – a partir do *prompt* –, ou *List* – selecionado do menu.

Este é um programa muito interessante de possuir, pois ele permite não só sua impressão, como também sua edição rápida e eficiente, além de direta.

Para usar o SideKick simultaneamente com o dBASE III Plus, após carregar o DOS, carregue o SideKick no mesmo acionador *antes* de carregar o dBASE. Insira o disquete do SideKick no acionador A e digite:

S I D E W A Y S Version 2.01
(C) Funk Software, Inc. 1982

Ser#3475318
E100

```

Vertical form size (inches):  8.00          Printer select:  LPT1:
Horizontal form size (inches): 8.00

Double strike:                OFF
Character font:                LARGE        7 x 15 dot matrix

Character spacing (dots):      2           8.00 characters per inch
Line spacing (dots):           3           6.66 lines per inch

Left margin (inches):         2.00
Max printing width (inches):  10.00        80 characters per line

Top margin (inches):          1.00
Printing length (inches):     6.00        40 lines per page

Enter name of print file:  B:CENTRAL.PRg   Starting page:      1
                                           Glue lines:        0

Strike "" and "" to move from field to field
      "F1" to save current options
      "F10" to exit from program

```

Figura 15.5: Menu do SIDEWAYS com central.prg.

SK ENTER

e você deverá ver uma tela mostrando o programa carregado na memória, como na Figura 15.7.

Observe que sua tela de abertura exibe a quantidade de memória disponível para todo o sistema antes do carregamento do dBASE e o *prompt* do DOS, liberando o sistema para o carregamento do dBASE.

A seguir, carregue o dBASE e abra o arquivo VENDAS.DBF via menu ou *prompt* (não se esqueça de acionar SET DEFAULT TO B caso você prefira o *prompt*).

Liste VENDAS.DBF com LIST ou List (ou seja, via *prompt* ou menu). Observe que ele contém 20 registros (se você não construiu este arquivo como pedimos há alguns capítulos atrás, faça isso agora, pois todos os 20 registros serão necessários para nosso trabalho com o SideKick).

```

***** Programa central.prg
***** Menu Central de Decisoes

set bell off
set status on
x = 0
do while x <> 5
clear
text
    MENU Principal
    (escolha um numero)

    1 = Agenda telefonica.
    2 = Etiquetas postais
    3 = Relatorio de vendas
    4 = Atualizacao
    5 = Saida
endtext
@ 10,15 say 'Escolha um numero: ' get x pictu
read
do case

    case x = 1
        do caso1
    case x = 2
        do caso2
    case x = 3
        do caso3
    case x = 4
        do caso4
endcase
enddo (while x <> 5)
set bell on
return

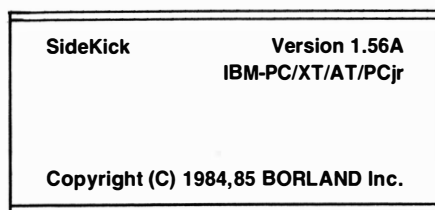
```

Figura 15.6: Resultado da impressão (parcial) de central.prg via SIDEWAYS.

Note que os títulos que representam os nomes de campo (CLIENTE, PRODUTO, QTDE etc.), não aparecem na exibição. Isto ocorre por causa do tamanho limitado da tela, obviamente.

Com o SideKick editaremos novos campos para esse arquivo, além de visualizá-lo e imprimi-lo totalmente.

Com o arquivo VENDAS.DBF listado em sua tela do dBASE, chame o SideKick por meio dos seguintes procedimentos:



Full System

720896 bytes total memory
 663856 bytes were free
 558496 bytes free

A >

Figura 15.7: Menu de abertura do Sidekick.

1) Ctrl-Alt (para chamar o menu do SideKick na tela de VENDAS.DBF). Sua tela agora deverá estar como na Figura 15.8.

2) F2 (para chamar o que o SideKick denomina “bloco de anotações” – o *NotePad* –, que será sua primeira área de trabalho no programa).

3) Nome do arquivo (digite o nome do arquivo do SideKick cujo papel será armazenar o arquivo que está na tela do dBASE. Aqui, sugerimos o nome SKVENDAS). Caso sua tela não esteja pronta para digitar o nome do arquivo, tente com F3. Sua tela com seu novo arquivo receptor SKVENDAS deverá estar como na Figura 15.9. (Qualquer problema durante seu trabalho, tecle Esc para voltar à tela do dBASE e comece tudo de novo.)

4) F4 (para “importar” os dados da tela atual do dBASE para sua tela atual do SideKick, para futura edição).

5) F7 (para marcar, com ENTER, o início e o fim da tela do dBASE que você deseja editar. Pode-se marcar e trabalhar com somente uma parte da tela, caso seja necessário. Para iniciar a marcação, simplesmente leve o cursor ao local desejado e tecle ENTER. Em seguida, leve o cursor para o fim do arquivo desejado – que pode ser somente parte da tela do dBASE e tecle ENTER para marcar – iluminando toda a área marcada – seu final).

6) Esc (para sair da tela demarcada por F7).

7) Ctrl-KC (para inicializar sua tela de edição. Nesta tela pode-se editar normalmente, alterando qualquer parte do arquivo. Desça a tela do dBASE de modo que você possa incluir novos títulos – com Ins – no lugar dos nomes dos campos, de acordo com a Figura 15.10).

Set Up	Create	Update	Position	Retrieve	Organize	Modify	Tools	01.59.41 am
1	001	LOTUS T.A.	40	4.00	07/20/88	.T.		
2	002	WordLegal	60	15.00	07/20/88	.T.		
3	003	WordLegal	20	15.00	07/20/88	.T.		
4	003	dBASE III	15	6.00	07/20/88	.T.		
5	001	APPLE	10	4.00	07/18/88	.T.		
6	010	ASMIBM PC	80	6.00	07/18/88	.T.		
7	042	ASMIBM PC	25	6.00	07/17/88	.T.		
8	009	LOTUS T.A.						
9	033	WordStar						
10	012	MKT						
11	005	MKT						
12	008	MATFIN						
13	014	WordLegal						
14	021	dBASE III						
15	022	dBASE III						
16	038	WordLegal						
17	037	ASMIBM PC						
18	001	dBASE III						
19	007	WordLegal	20	4.00	06/06/88	.F.		
20	019	WordLegal	60	6.00	06/04/88	.F.		

SideKick Main Menu
F1 Help
F2 NotePad
F3 Calculator
F4 Calendar
F5 Dialer
F6 Ascii-table
F7 Setup
Esc exit

ASSIST	<B:>	VENDAS	Rec: EOF/20
--------	------	--------	-------------

Press any key to continue work in ASSIST

↑↓ move bar. Select by pressing a highlighted letter, a function key, or ↵

Figura 15.8: Menu do SideKick sobre VENDAS.DBF.

8) F2 (para salvar todo o seu trabalho).

9) Ctrl-KP (para imprimir seu trabalho, isto é, sua nova tela de VENDAS.DBF).

Sua impressão de SKVENDAS, ou seja, do novo arquivo VENDAS.DBF editado pelo SideKick, deverá estar como na Figura 15.10.

Lembre-se, qualquer problema durante essa operação, tecle Esc para começar tudo de novo.

INTERFACE dBASE III Plus – Turbo Basic

O editor de textos do Turbo Basic também pode aceitar arquivos do dBASE III Plus para posterior edição. Como exemplo, trabalharemos com central.prg, transferindo-o para a tela do editor do Turbo Basic (pode-se também enviar outros tipos de arquivos, como os arquivos .TXT, para o editor do Turbo Basic).

Para iniciar, carregue o Turbo Basic após o MS-DOS (ou o PC DOS), digitando, com o disquete do Turbo Basic no acionador A:

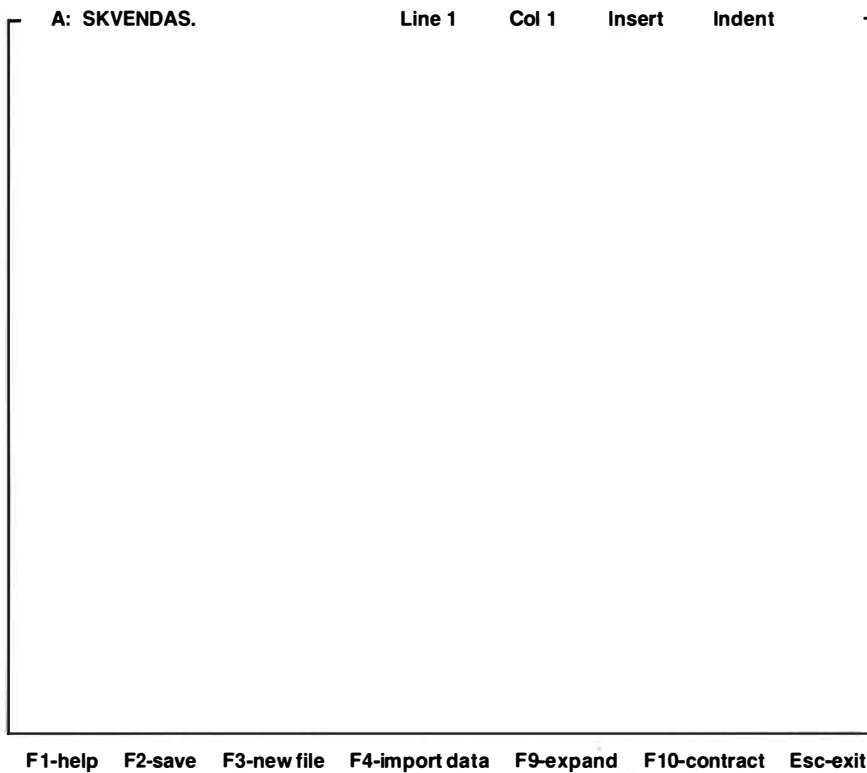


Figura 15.9: Tela do SideKick com SKVENDAS.

TB ENTER

Sua tela deverá estar como na Figura 15.11.

Para chamar seu arquivo do dBASE no acionador B, selecione a opção *File* do Turbo Basic. A seguir, selecione *Load* (Figura 15.12), o nome do acionador (B:), digite o nome do programa (central.prg) e pressione ENTER.

Tecele F5 (*Zoom*) para ver o programa inteiro na tela.

(Ainda não foi suficiente, vá para a opção *Edit* e tecele ENTER). Sua tela parcial, já com central.prg, deverá estar como na Figura 15.13 e sua tela final, como na Figura 15.14. Agora é só trabalhar à vontade!

Cliente	Produto	Qtde.	OTN.	Datas	Cond
001	LOTUS T.A.	40	4.00	07/20/88	.T.
002	WordLegal	60	15.00	07/20/88	.T.
003	WordLegal	20	15.00	07/20/88	.T.
003	dBASE III	15	6.00	07/20/88	.T.
001	APPLE	10	4.00	07/18/88	.T.
010	ASMIBM PC	80	6.00	07/18/88	.T.
042	ASMIBM PC	25	6.00	07/17/88	.T.
009	LOTUS T.A.	8	4.00	07/12/88	.T.
033	WordStar	80	4.00	07/08/88	.T.
012	MKT	80	4.00	06/30/88	.F.
005	MKT	10	4.00	06/30/88	.F.
008	MATFIN	80	4.00	06/22/88	.F.
014	WordLegal	40	15.00	06/22/88	.F.
021	dBASE III	20	6.00	06/18/88	.F.
022	dBASE III	10	6.00	06/18/88	.F.
038	WordLegal	30	6.00	06/12/88	.F.
037	ASMIBM PC	5	6.00	06/08/88	.F.
001	dBASE III	80	6.00	06/08/88	.F.
007	WordLegal	20	4.00	06/06/88	.F.
019	WordLegal	60	6.00	06/04/88	.F.

Figura 15.10: VENDAS.DBF editado com o SideKick.

INTERFACE dBASE III Plus – Lotus 1-2-3

O Lotus 1-2-3 precisa de arquivos do tipo .WKS para poder trabalhar. E com o dBASE III Plus pode-se transformar arquivos do tipo .DBF, por exemplo, em arquivos do tipo .WKS, com muita facilidade. Como exemplo, vamos transformar o arquivo VENDAS.DBF num arquivo especial para ser lido e trabalhado pelo Lotus. Para isso, digite, ainda no dBASE:

USE VENDAS ENTER

para abrir o arquivo VENDAS. Em seguida, digite:

COPY TO B:LVENDAS TYPE WKS ENTER

para transformar este arquivo num novo arquivo com formato do tipo .WKS agora chamado de LVENDAS (de Lotus + VENDAS).

Agora, seu arquivo acabou de ser copiado para um novo arquivo, LVENDAS, que tem o formato específico de leitura do Lotus 1-2-3.

Saia do dBASE com QUIT e carregue seu Lotus no acionador A com:

123 ENTER

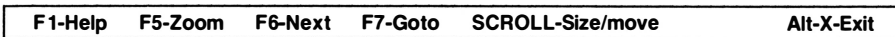
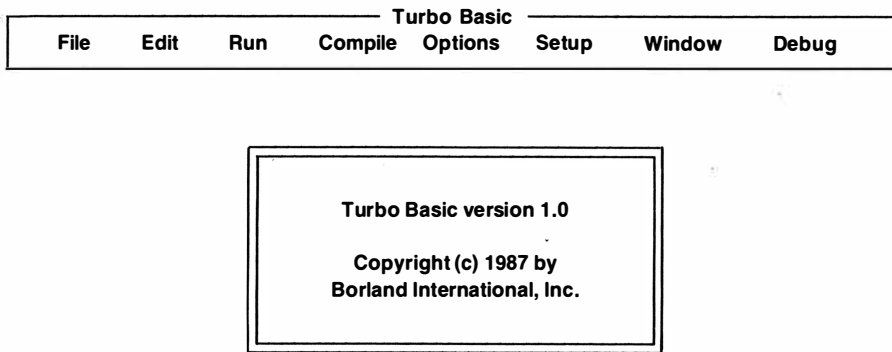


Figura 15.11: Tela inicial do Turbo Basic.

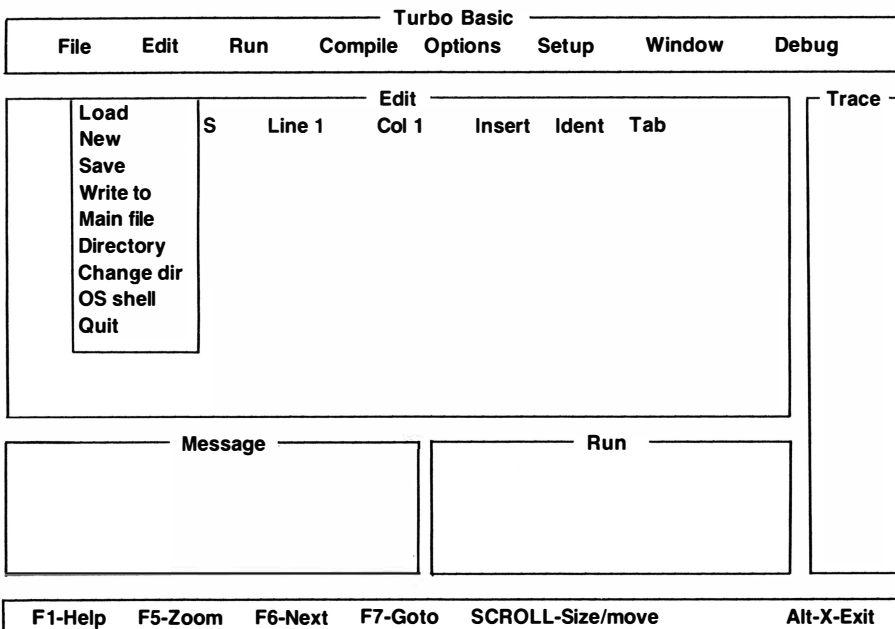


Figura 15.12.

Deixe o disquete de trabalho do dBASE no acionador B e chame o arquivo LVENDAS.WKS com /B:LVENDAS. Observe que LVENDAS não “entra” no Lotus “exatamente” como está no dBASE. Há algumas alterações de formato de campos, como por exemplo, a transformação do campo VENC (as datas encontram-se num formato diferente)

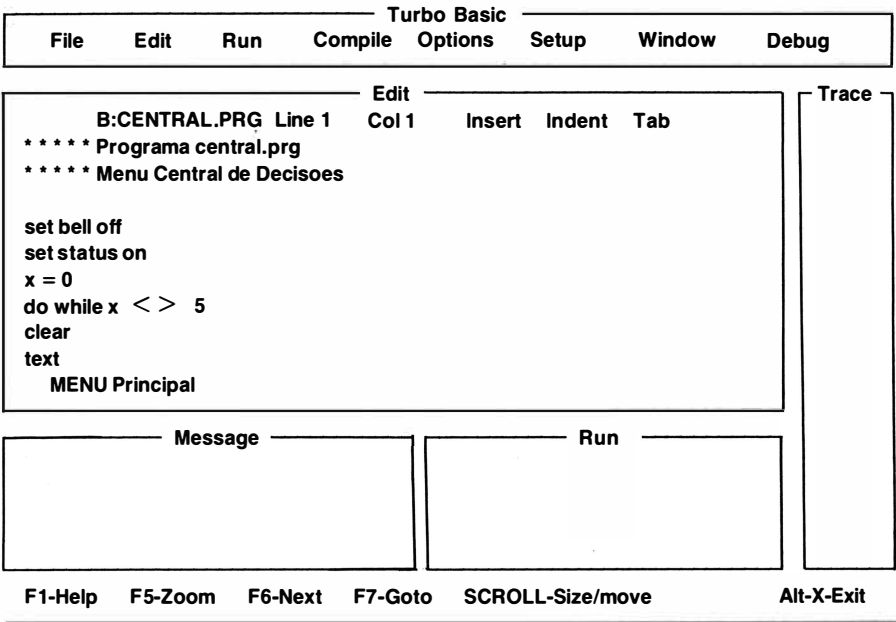


Figura 15.13.

```

B:CENTRAL.PRG Line 1   Col 1   Insert   Indent   Tab
***** Programa central.prg.
***** Menu Central de Decisooes

set bell off
set status on
x = 0
do while x <> 5
clear
text
    MENU Principal
    (escolha um numero)

    1 = Agenda telefonica
    2 = Etiquetas postais
    3 = Relatorio de vendas
    4 = Atualizacao
    5 = Saida
endtext
@ 10,15 say 'Escolha um numero: ' get x picture '9' range 1,5
read
do case

    case x = 1

```

Figura 15.14.

e a união desse campo com o campo STATUS. Note também que, como estamos no Lotus, o registro número um começa com os nomes de campo do dBASE. Veja tudo na Figura 15.15.

```

A1: [W4] CLIENTE                                READY

      A      B      C      D      E      F      G      H      I      J
1  CLIEPRODUTO  QTDOTN  VENC  STATUS
2  001 LOTUS T.A. 40      4 19880720T
3  002 WordLegal 60      15 19880720T
4  003 WordLegal 20      15 19880720T
5  003 dBASE III 15      6 19880720T
6  001 APPLE     10      4 19880718T
7  010 ASMI8M PC 80      6 19880718T
8  042 ASMI8M PC 25      6 19880717T
9  009 LOTUS T.A. 8      4 19880712T
10 033 WordStar 80      4 19880708T
11 012 MKT       80      4 19880630F
12 005 MKT       10      4 19880630F
13 008 MATFIN    80      4 19880622F
14 014 WordLegal 40      15 19880622F
15 021 dBASE III 20      6 19880618F
16 022 dBASE III 10      6 19880618F
17 038 WordLegal 30      6 19880612F
18 037 ASMI8M PC 5       6 19880608F
19 001 dBASE III 80      6 19880608F
20 007 WordLegal 20      4 19880606F
20-Jul-88 11:58 AM

```

Figura 15.15.

Agora você pode efetuar suas modificações editoriais e seu arquivo estará pronto para ser trabalhado tanto no dBASE (como VENDAS.DBF – que ainda permanece no disquete) como no Lotus (via LVENDAS – que se encontra no mesmo disquete e que pode ser transferido para um disquete específico do Lotus).

Transferência Lotus – dBASE

Transferir arquivos feitos no Lotus para o dBASE III Plus também é uma tarefa relativamente fácil. Vejamos seu procedimento.

Chame a opção **Worksheet, Column-Width, Set**. Altere o tamanho do campo A1 para 15 caracteres, inserindo o número 15 ou usando a seta para a direita até o ponto desejado. Pressione ENTER quando chegar ao tamanho de 15 caracteres para confirmá-lo. Faça o mesmo com os campos B1 (para 25 caracteres) e C1 (também para 25 caracteres). Nas posições A1, B1 e C1, insira as palavras *Nome*, *Sobrenome* e *Endereço*, respectivamente. Na célula A3, insira o nome do autor deste livro, Roger (ou qualquer outro nome que você queira) e na célula B3 seu sobrenome, Dalton. Na célula C3, insira a palavra Berkeley, como exemplo de endereço para Roger Dalton. Vá para a célula A4 e insira Gisella. Na célula B4, coloque o sobrenome, Narcisi e na célula C4 a cidade de São Paulo, como exemplo de endereço. Em seguida, vá para a célula A5 e insira o nome do editor desta obra, Ricardo. Na célula B5, coloque seu sobrenome, Reinprecht. Na célula C5, insira a cidade do Rio de Janeiro, como exemplo de endereço. A seguir, vá para a célula A6 e coloque o nome do Gerente de Vendas (se precisar de alguma coisa desta editora, *ele é o homem!*), Wilson. Na célula B6 insira seu sobrenome com a inicial de seu nome do meio, R. Paulino. Finalmente, na célula C6, coloque o nome da cidade de São Paulo, como exemplo de endereço. Sua tela do Lotus 1-2-3 com o pequeno arquivo criado como exemplo de entrada no dBASE III Plus deverá se parecer com a tela da Figura 15.16.

A16: [W15]

READY

	A	B	C
1	Nome	Sobrenome	Endereco
2			
3	Roger	Dalton	Berkeley
4	Gisella	Narcisi	Sao Paulo
5	Ricardo	Reinprecht	Rio de Janeiro
6	Wilson	R. Paulino	Sao Paulo
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
22-Jul-88 08:52 PM			

Figura 15.16: Arquivo inicial em Lotus 1-2-3.

Grave seu arquivo na memória sob o nome TRANSFDB, com os comandos */File, Save, nome do arquivo*. Saia do Lotus com */Quit* e carregue o dBASE.

Para converter seu arquivo do Lotus 1-2-3 para o dBASE III Plus e anexá-lo a um arquivo existente no dBASE, primeiro você deve ter uma estrutura de arquivo (ou seja, um arquivo!) igual ou maior que a estrutura do arquivo do Lotus. Isto quer dizer que nossa estrutura deve ter no mínimo um tamanho de 15 caracteres para o primeiro campo, 25 para o segundo e 25 para o terceiro (porém, a estrutura receptora do dBASE pode ser menor, desde que o conteúdo do campo do arquivo Lotus seja menor ou igual. Por exemplo, se o campo TEL de seu arquivo CLIENTES.DBF tiver um tamanho de 14 caracteres, a palavra Rio de Janeiro caberá inteira no arquivo dBASE, mas se este mesmo campo tiver 12 caracteres, então o nome da cidade ficará incompleto como "Rio de Janei"). Decidimos então usar o arquivo CLIENTES.DBF para "puxar" o arquivo TRANSFDB do Lotus 1-2-3 (se você não possuir um arquivo com este tipo de estrutura, crie um agora, antes da conversão!).

Com o dBASE carregado no acionador A e seu disquete com o arquivo TRANSFDB criado no Lotus 1-2-3 no acionador B:, digite:

SET DEFAULT TO B ENTER

e, a seguir,

USE CLIENTES ENTER

(ou, se preferir, use o menu para abrir CLIENTES.DBF com *Set Up, Database file, B:, CLIENTES.DBF, N* ou *Y*).

Após abrir CLIENTES.DBF ou outro arquivo com estrutura semelhante, no mínimo do mesmo tamanho, digite:

APPEND FROM B:TRANSFDB.WK1 TYPE WKS ENTER

Ao terminar esta operação TRANSFDB.WK1 já estará unido ao arquivo CLIENTES.DBF do dBASE. Para conferir, liste CLIENTES na tela e você verá seu novo arquivo como na Figura 15.17 (talvez seu arquivo CLIENTES.DBF não esteja como o nosso, mas não se preocupe com isso).

Observe que TRANSFDB.WK1 fica logo abaixo de CLIENTES.DBF, sendo que o campo Endereço é parte integrante do registro 6 juntamente com os campos Nome e Sobrenome. Isto aconteceu, obviamente, porque nosso arquivo receptor só continha 5 registros e a união dos arquivos acontece exatamente a partir do próximo registro vazio. Agora, é só editar seu novo arquivo e começar a trabalhar!

Record#	SNOME	NOME	TEL
1	Narcisi	Gisella	
2	Pinduba	Larissa	
3	Silva	Joaquim	
4	Silva	Joao	
5	SILVA	Jose	
6	Nome	Sobrenome	Endereco
7	Roger	Dalton	Berkeley
8	Gisella	Narcisi	Sao Paulo
9	Ricardo	Reinprecht	Rio de Janeiro
10	Wilson	R. Paulino	Sao Paulo

Figura 15.17: TRANSFDB.WK1 unido a CLIENTES.DBF.

CONVERSÃO dBASE III Plus – WordStar

Aqui faremos nossos comentários mais demorados, pois consideramos o WordStar o maior aliado do dBASE para colocar em prática todo um sistema de endereçamento, incluindo cartas padronizadas – e personalizadas – e etiquetas de endereçamento, para auxiliar e aproveitar ao máximo os arquivos criados no dBASE, bem como para editar relatórios comerciais que podem permanecer no WordStar para serem impressos em séries personalizadas. Também criaremos um espaço específico (um arquivo) para imprimir o arquivo TELA.TXT, preparado no capítulo anterior, que poderá ser fornecido por um disquete do WordStar ou do próprio dBASE, onde ele se encontra originalmente. Os relatórios comerciais de que falamos há pouco serão representados pelo arquivo VENDAS.FRM – o relatório especial do arquivo VENDAS.DBF criado anteriormente.

Imprimindo TELA.TXT

Crie um novo arquivo no dBASE, com CREATE ou *Create*, sob o nome RECEPTEXT (RECEPTEXT.DBF). Entre no modo de inserção de campos e insira somente um campo com o nome RECEPTEXT, ou qualquer outro nome e programe sua estrutura para *Character*, com um tamanho de 80 caracteres (novamente, se você programar menos de 80 caracteres no total, sua impressão será parcial, reduzida apenas à largura programada pelo arquivo receptor).

Sua tela ficará truncada devido ao tamanho do arquivo, porém, se você programar sua impressora para o modo de caracteres comprimidos (em nosso exemplo, usamos uma Grafix GS1000, da Scritta Eletrônica Ltda., com a chave de configuração 3 do Conjunto 1 em ON), sua impressão final deverá ficar como a Figura 15.18.

```

Record# RECEPTEXT
1 Field definitions for Screen : B:TELA.scr
2
3 Page Row Col Data Base Field Type Width Dec
4 1 6 12 LIVRARIA NOME Character 40
5 1 8 12 LIVRARIA ENDereco Character 40
6 1 10 12 LIVRARIA CIDADE Character 15
7 1 10 37 LIVRARIA ESTADO Character 2
8 FUNCTION !
9 1 10 46 LIVRARIA CEP Character 6
10 1 14 12 LIVRARIA TEL Character 14
11 PICTURE (999)999-9999
12 1 16 0 LIVRARIA REM Memo 10
13
14
15 Content of page : 1
16
17 Banco de Dados - DISTRIBUIDORES/BRASIL
18
19
20
21
22 Nome XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX Auxilio na Tela
23
24 Endereco XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX Cursor: Setas
25 Paginas: PgDn e PgUp
26 Cidade XXXXXXXXXXXXXXX Estado XX Cep XXXXXX
27 Apagar: Letra = Del
28 Campo = Ctrl-Y
29 Registro = Ctrl-U
30 Tel XXXXXXXXXXXXXXX
31 Memo: Ctrl-PgDn(dentro)
32 XXXX
33 Salvar: Ctrl-End
34 Desistir: Esc

```

Figura 15.18: RECEPTEXT.DBF com TELA.TXT.

Convertendo VENDAS.FMT

Para gerar o relatório de vendas do arquivo VENDAS.DBF diretamente num formato .TXT para edição no WordStar, digite, enquanto no dBASE:

```
USE VENDAS    ENTER
REPORT FORM VENDAS TO RELTEXTO    ENTER
QUIT    ENTER
```

para converter seu relatório no formato .TXT. (Use **MODIFY REPORT** *nome do arquivo*, para preparar o relatório, caso ainda não o tenha feito.)

Em seguida, saia do dBASE, após certificar-se de que o seu arquivo está gravado no disquete do acionador B (ou qualquer outro meio de armazenamento) e carregue o WordStar, digitando WS junto ao *prompt* do DOS.

Quando o Menu de Abertura do WordStar aparecer, tecle D para determinar que você quer a opção *Document*. Na posição *open a document*, insira o nome do arquivo, como no exemplo:

```
B:RELTEXTO.TXT    ENTER
```

ou Ctrl-KR, caso você já esteja dentro de um arquivo no WordStar e queira inserir seu arquivo convertido do dBASE na posição do cursor.

Bem, nessa altura VENDAS.FMT, ou seja, agora RELTEXTO.TXT, já está no seu programa WordStar.

Um Sistema de Endereçamento Completo com o Merge

Vamos aproveitar que já construímos um sistema duplo de etiquetas de endereçamento no dBASE, para transferir seu arquivo-fonte LIVRARIA.DBF para o WordStar e em seguida, via *Merge Print* (que era chamado de *MailMerge* nas versões anteriores do WordStar), construir um sistema de endereçamento de envelopes, cartas personalizadas e, “de quebra”, um sistema de etiquetas de endereçamento direto para funcionar automaticamente com a emissão dessas cartas.

Com o dBASE ativo, digite:

```
USE LIVRARIA    ENTER
```

para abrir seu arquivo-fonte de endereços. Para converter este arquivo (do tipo .DBF) para o formato .TXT, acione o comando:

```
COPY TO CARTA DELIMITED    ENTER
```

Crie, em seguida, um arquivo de comandos no WordStar, com a opção D. Dê a este arquivo o nome de ENVCARTA.MD (de *Meu Diretório*), o arquivo principal de controle. Ao receber a tela vazia de ENVCARTA, insira os comandos, exatamente como sugerimos:

```
.FI WSCARTA.MD  ENTER
(pule esta linha com ENTER)
.FI WSENVLP.MD ENTER
```

Não se esqueça de teclar ENTER após a última linha!

Retorne para o menu com Ctrl-KD para criar o primeiro arquivo. Selecione D novamente e digite B:WSCARTA.MD como o nome do arquivo.

Copie as instruções abaixo exatamente como na Figura 15.19. Este programa será nossa carta personalizada – parte do sistema mais amplo que vem por aí.

```
.OP
.MT 3
.DF B:CARTA.TXT
.RV NOME,ENDEREÇO,CIDADE,ESTADO,CEP,TEL
```

```
%NOME%
%ENDEREÇO%
%CIDADE%, %ESTADO%
%CEP%
```

Prezados Senhores:

Tendo em vista o interesse de vossa conceituada empresa, %NOME%, vimos por meio desta, comunicar que, dentro em breve, colocaremos no mercado mais uma publica,^Hca^H~o sobre ^BdBASE III Plus^B, de ^BRoger Dalton^B - o mais recente "Best Seller" internacional!

Mais uma vez, obrigado pela aten,^Hca^H~o.

Cordialmente,

O Editor.

```
.PA
```

Figura 15.19: Programa WSCARTA.MD

Note que os símbolos especiais representam os sinais de acentuação para a língua portuguesa. Os “&” representam as variáveis de seu programa, que originar-se-ão no programa-fonte CARTA.TXT (que por sua vez veio do dBASE).

Salve seu arquivo com Ctrl-KD e abra outro com a opção D, chamado WSENVLP.MD, que será usado para a produção automática de etiquetas de endereçamento e copie-o de acordo com a Figura 15.20.

Aqui, repetimos a linha de comando .RV, com números para variáveis, para que o programa entenda que imprimiremos duas vezes, ou seja duas colunas, uma ao lado da outra, o que também é confirmado pelo restante do programa. A linha de comando .SV anterior, revela o tamanho máximo de caracteres (40) que queremos aceitar em cada coluna do formulário contínuo, justificando-os, simultaneamente, à esquerda (note que usamos a letra “L” de *Left*. Podemos optar ainda entre as letras “R” e “C”, para “direita” e “centralizado”, respectivamente).

```
.OP
.PF OFF
.PL 4
.MT 0
.MB 0
.PD 0
.SV L = LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL
.DF B: CARTA.TXT
.RV NOME1, ENDereco1, CIDADE1, ESTADO1, CEP1, TEL
.RV NOME2, ENDereco2, CIDADE2, ESTADO2, CEP2, TEL

.SV CIDADESTADO1 = &CIDADE1& &ESTADO1&
.SV CIDADESTADO2 = &CIDADE2& &ESTADO2&
&NOME1/L&      &NOME2/L&
&ENDereco1/L&   &ENDereco2/L&
&CIDADESTADO1/L& &CIDADESTADO2/L&
&CEP1/L&        &CEP2/L&
```

Figura 15.20: Programa WSENVLP.MD.

Usando o Sistema Completo de Endereçamento

Para usar seu sistema, certifique-se de que os quatro arquivos de que tratamos até aqui (CARTA.TXT, ENVCARTA.MD, WSCARTA.MD, WSENVLP.MD) estejam no mesmo disco. Salve tudo com Ctrl-KD e chame seu arquivo de controle, ENVCARTA.MD (com a opção M de *Merge print*), que se encarregará de chamar seu arquivo de cartas personalizadas primeiro (acessando WSCARTA) e depois, automaticamente, chamará WSENVLP, nosso arquivo de etiquetas.

Endereçamento Direto no Envelope

Pode-se endereçar o envelope diretamente e evitar uma sucessão de etiquetas inúteis para determinada tarefa. Isto é o que faremos agora.

Copie as instruções do Programa WSENDENV.MD, exatamente de acordo com a Figura 15.21.

```
.DF
.PO 10
.DF B:CARTA.TXT
.RV NOME, ENDERECO, CIDADE, ESTADO, CEP, TEL
.CS
.DM Coloque o envelope corretamente
.DM      (Cancelar: Ctrl-U)
.DM
.DM
.DM
.DM
^C

&NOME&
&ENDERECO&
&CIDADE&, &ESTADO&
&CEP&
```

Figura 15.21: Programa WSENDENV.MD de endereçamento direto.

Observe que $\hat{C}$ é feito pressionando-se Ctrl-PC. Salve seu programa com Ctrl-KD. Para usar todo o sistema plenamente, sugerimos que você o coloque num sistema com programação condicional, para escolher se deseja envelopes únicos ou etiquetas de endereçamento (o que também pode ser feito com as cartas), ou, caso seu conhecimento do WordStar ainda não seja suficiente (neste caso procure nossa próxima publicação, WordStar Professional 4), você poderá inserí-lo num sistema idêntico ao sistema ENVCARTA.MD, para que ele tenha acesso direto também às cartas.

Se você quiser usá-lo imediatamente, tecle M no menu de abertura do WordStar e responda (se quiser) às perguntas do programa. Certifique-se de responder com “N” à pergunta que precede a solicitação do tipo da impressora.

Etiquetas de Uma Coluna

Como complemento final, suponha que você queira programar etiquetas para serem impressas em uma só coluna de formulário contínuo. Para isso, copie o programa da Figura 15.22 e depois compare-o com WSCARTA.MD e repare as diferenças entre os dois programas.

```
. OF
. PF OFF
. PL 6
. MT 0
. MB 0
. PD 0
. DF B: CARTA.TXT
. RV NOME, ENDERECO, CIDADE, ESTADO, CEP, TEL
```

```
&NOME%
&ENDERECO%
&CIDADE%, &ESTADO%
&CEP%
```

Figura 15.22: Programa de etiquetas de uma coluna.

FORMATOS PADRÃO PARA TRANSFERÊNCIAS DE ARQUIVO

São esses os formatos padrão para transferência de arquivos entre o dBASE III Plus e outros *softwares*:

ASCII
DELIMITED
DIF
SDF
SYLK
WKS

além da forma simplificada com os arquivos do tipo .TXT ou qualquer outro que poderá ser lido pelo WordStar com Ctrl-KR.

O formato ASCII já foi usado anteriormente com o comando SET ALTERNATE. Este formato também pode ser usado, juntamente com os formatos SDF e DELIMITED, com outros processadores de texto, como o Microsoft Word® ou o MultiMate®. Neste capítulo já usamos o formato DELIMITED para transformar o arquivo LIVRARIA.DBF no arquivo de texto CARTA.TXT. Este formato também serve para interfaces com outros gerenciadores de bancos de dados, como o R:base 5000®, por exemplo.

Um exemplo com arquivos DIF seria a transferência de um arquivo dBASE para um arquivo VisiCalc®. Isto seria feito com:

COPY TO B:NOMEARQ TYPE DIF ENTER

onde B:NOMEARQ significa o acionador e o nome do arquivo receptor. Lembre-se também de que seu arquivo fonte deverá estar aberto (USE) para que o comando COPY saiba de onde copiar.

Já usamos a extensão .SDF há pouco com o WordStar. Passemos então para o próximo exemplo.

Para transferir arquivos do tipo SYLK (bem como para todos os outros, como você já deve ter notado), isto é, arquivos dBASE para arquivos Multiplan®, digite:

COPY TO B:NOMEARQ TYPE SYLK ENTER

Para transferir dos programas externos para o dBASE, use os mesmos procedimentos, como no exemplo a seguir:

APPEND FROM B:NOMEARQ TYPE SYLK

e assim por diante.

Usando IMPORT e EXPORT

Os comandos IMPORT e EXPORT, também encontrados no menu do dBASE sob a opção *Tools*, é usado exclusivamente para transferências entre o dBASE e arquivos do programa *PFS File*, com extensão do tipo .PFS.

Além do menu, pode-se também usar os comandos do *prompt*, como nos seguintes exemplos:

```
IMPORT FROM B:NOMEARQ TYPE PFS  ENTER
```

para ler um arquivo desse tipo no dBASE. Para “exportar” um arquivo dBASE para o *PFS File*, digite:

```
EXPORT TO B:NOMEARQ TYPE PFS  ENTER
```

onde B:NOMEARQ agora é o nome do arquivo receptor.

CONVERSÃO dBASE II – dBASE III Plus

Como prometemos no início do capítulo, para finalizar, vamos trabalhar com a conversão de arquivos dBASE II para dBASE III Plus.

Acesse seu programa dCONVERT, do dBASE III Plus. Para inicializá-lo, digite:

```
dCONVERT A: B:  ENTER
```

ou A: C:, caso seu sistema seja de disco rígido.

Note que você recebe um menu de tipos de conversão, como na Figura 15.23, para que você possa escolher entre programas .DBF, ou de outros tipos de extensão, como .MEM, .FRM, .PRG, .FMT, e, finalmente, .NDX. Note também que a opção número 7 desse mesmo menu, também oferece uma possibilidade de conversão dBASE III Plus-dBASE II, para o tipo .DBF, contanto que as regras de tamanhos de registro e quantidades de campos, entre outros limites, sejam observadas

Conversões em Lote

A partir do menu, é muito fácil operacionalizar essas conversões, porém, pode-se digitar comandos diretamente do *prompt* para efetuar conversões. Por exemplo, para converter um arquivo do tipo CENTRAL.PRG do dBASE II para o dBASE III Plus diretamente do DOS, via arquivos em lote, digite (diretamente no *prompt* do DOS):

dBASE CONVERT – dBASE III File Conversion Aid v2.0 08/14/85
(c) 1984 By Ashton-Tate All Right Reserved

dBASE II -> dBASE III

1 – Database File	<.DBF>
2 – Memory Variable File	<.MEM>
3 – Report Format File	<.FRM>
4 – Command File	<.PRG>
5 – Screen Format File	<.FMT>
6 – Index File Help	<.NDX>
7 – Un-dCONVERT III -> II	<.DBF>
9 –	Instructions
0 –	Exit

< Use cursor arrows to move between choices: hit RETURN to select choice >

Figura 15.23: Menu de dCONVERT.

DCONVERT*.PRG ENTER

e *todos* os arquivos do dBASE II que estiverem no formato .PRG serão convertidos.

Você notará que os programas convertidos nem sempre ficam exatamente como eram no dBASE II. Mas lembre-se, é bem mais fácil editar ou efetuar pequenas correções do que ter de escrever todos os programas novamente no dBASE III. A forma mais indicada para analisar sua documentação convertida é fazer uma cópia impressa de seus programas para apontar as pequenas diferenças entre o que se apresenta como resultado da conversão e o que o dBASE III Plus efetivamente aceita em sua lógica.

Esperamos ter definitivamente colaborado para seu melhor entendimento das possibilidades de interface entre o dBASE III Plus e diversos outros *softwares* disponíveis no mercado. Acreditamos que você, o usuário, saiba, de agora em diante, aplicar os exemplos apresentados (e testados) neste livro em seus próprios programas ou para as suas próprias necessidades.

No próximo capítulo mostraremos como compilar seus programas com o CLIPPER, de modo que você possa rodá-los diretamente do DOS, sem a intermediação do dBASE III. No Capítulo 17 - LAN - Redes Locais com o dBASE III Plus, abordaremos os conceitos básicos dessa prática bem como as necessidades básicas para a instalação de um sistema em rede.

Capítulo

16

Clipper **Compilando Sem Erros**

Neste capítulo pretendemos mostrar como compilar todos os programas do livro via SuperMacro central.prg. Faremos uma compilação *real* e não uma simples abordagem, como a maioria dos livros sobre o assunto.

Quando você terminar a sessão de programação, compilação e “linkagem”, estará apto a compilar qualquer programa adequado para esta tarefa.

Mostraremos também, algumas adaptações no sistema central.prg, para que você entenda melhor alguns detalhes importantes – e fundamentais – sobre a adequação de programas para posterior compilação. Iniciaremos o processo de compilação após a análise dos programas adaptados.

CONHECIMENTOS BÁSICOS DO PROGRAMADOR

Se você pretende compilar um programa, é preciso que saiba primeiro programar em dBASE. E isto significa conhecer detalhes básicos da programação do tipo .PRG – os programas preparados via comando MODIFY COMMAND.

Para o nosso caso, o pré-requisito será o conhecimento total dos sistemas .PRG feito no livro, primeiro com a introdução do SuperMacro auto@.prg e depois com o detalhamento do sistema central.prg. Conhecidas suas técnicas, estaremos com cerca de 70% do caminho andado para a chegada final, que será a transformação do sistema central.prg em CENTRAL.EXE – totalmente executável via DOS e, o mais importante, com o fonte inacessível para o usuário! (Também incluímos CENTRAL.EXE no disquete opcional, disponível em sua livreria de preferência).

FERRAMENTAS BÁSICAS PARA COMPILAÇÃO

Para compilar um programa com CLIPPER, precisamos no mínimo dos arquivos CLIPPER.EXE, CLIPPER.LIB e PLINK86 (ou outro tipo de “linkeditor”). Você pode ter esses arquivos (e os outros que fazem parte do programa) armazenados em seu disco rígido ou separados em dois disquetes com CLIPPER.EXE em um disquete e CLIPPER.LIB e PLINK86 juntos em outro disquete.

Compilando em Disquetes Flexíveis

O Processo de compilação em disquetes flexíveis pode se tornar em uma tremenda dor-de-cabeça, caso você não atente para alguns fatos importantes. Aqui não pretendemos escrever outro livro, que é o que teríamos de fazer se quiséssemos ilustrar todas (ou “quase todas”) as características do CLIPPER. Por isso, não abordaremos especificamente dois pontos, considerados dispensáveis do ponto de vista do tamanho e da proposta de nosso livro (o que certamente nos leva ao seu preço final): as interações com o programa CLIP-

PER e o uso do disco rígido. O primeiro porque só desejamos falar do processo de compilação e transformação em arquivos executáveis pelo DOS; o segundo, porque quem conhece o funcionamento do disco rígido, teoricamente não deverá ter nenhum problema em entender as interações com os disquetes do tipo *floppy* e, conseqüentemente, seus *drives*. E também porque, neste último caso, é mais difícil para o usuário/aprendiz lidar com dois acionadores numa compilação do que com um *Winchester*.

Os Acionadores de Disquetes

Para nossa compilação, serão necessários dois acionadores. Um para receber o executável (de preferência um disquete formatado vazio) e outro para o processo de “linkagem”, que conterá, simultaneamente, os arquivos PLINK86 (a versão utilizada pelo autor), CLIPPER.LIB (a biblioteca de auxílio do PLINK) e o programa .OBJ.

A Capacidade dos Disquetes

Certifique-se que seus disquetes estejam rigorosamente preenchidos somente com os arquivos necessários para a compilação. Retire quaisquer outros arquivos do disquete que contêm o PLINK86 e o CLIPPER.LIB. Esta é a melhor garantia de que você conseguirá uma boa e rápida compilação. Da mesma forma deixe somente os arquivos-fonte do dBASE e os programas .PRG no disquete que fará a primeira parte do trabalho – a compilação propriamente dita – e o disquete com o compilador CLIPPER.EXE também vazio, para receber seu programa objeto após esta primeira parte.

Central.Prg: Comandos Inválidos no CLIPPER

Copie novamente (ou simplesmente reforme) todos os programas do SuperMacro central.prg e observe as alterações que efetuamos em alguns programas. Para melhorar ainda mais seu estudo, aconselhamos que você siga exatamente nossas instruções, pois compilar um programa feito originalmente para funcionar somente com o dBASE III Plus – e não com o DOS –, não é tarefa fácil! Portanto, sugerimos agora que você faça uma cópia impressa de cada programa .PRG do sistema central.prg com o comando TYPE NOMEARQ.PRG TO PRINT (onde NOMEARQ representa o nome do programa) para que você tenha nas mãos toda a documentação necessária para a comparação (e posterior conclusão) com a versão adaptada para compilação.

Após a cópia de todos os programas, consulte a seção *Analisando os Procedimentos*, onde analisaremos todas as alterações efetuadas. Indicaremos os programas adaptados, isto é, os programas do Supermacro central.prg que foram modificados, ainda que pouco, com a marcação “/CLIPPER”. Após copiar cada programa (caso você prefira reformatar a versão já feita, faça uma cópia de reserva ou a cópia impressa antes) tire também uma cópia impressa de cada um. Observe a marcação “/CLIPPER” que colocamos junto ao cabeçalho de cada programa. Isto servirá para ajudá-lo nos seus estudos comparativos.

```
***** Programa central.prg/CLIPPER
***** Menu Central de Decisões
```

```
set bell off
set status off
x = 0
do while x < > 5
clear
text
    MENU Principal
    (escolha um numero)

    1 = Agenda telefonica
    2 = Etiquetas postais
    3 = Relatorio de vendas
    4 = Atualizacao
    5 = Saida
endtext
@ 10,15 say 'Escolha um numero: ';
get x picture '9' range 1,5
read
do case

    case x = 1
        do caso1
    case x = 2
        do caso2
    case x = 3
        do caso3
    case x = 4
        do caso4
endcase
enddo
set bell on
return
```

Programa central.prg./CLIPPER.

```
***** Programa CASO1.PRG/CLIPPER
***** Opera com o arquivo CLIENTES.DBF
```

```
set talk off
y = space(1)
clear
@ 10, 15 say 'Escolha um numero. ' +;
    'Relatorio = 1. Tela = 2: 'get y
read
if y = '2'
    use clientes
    clear
    list nome, snome, tel
    wait 'Pressione qualquer tecla para retornar' +;
        'ao menu anterior' to tecla
clear
```

```

endif
if y = '1'
    use clientes
    clear
    list nome, snome, tel to print
    wait 'Pressione qualquer tecla para retornar ' +;
    'ao menu anterior' to tecla
clear
endif
set talk on
return

```

Programa CASO1.PRG/CLIPPER.

```

***** Programa CASO2.PRG/CLIPPER
***** Submenu de etiquetas de enderecamento

```

```

R = space(1)
clear
@ 10,15 say 'Tipo de impressao: Global = 1. ' +;
'Especifica = 2: 'get R
read
if R = '2'
    do caso2A
endif
if R = '1'
    do caso2B
endif
return

```

Programa CASO2.PRG/CLIPPER.

```

***** Programa CASO2A.PRG/CLIPPER
***** Opera com o arquivo LIVRARIA.DBF

```

```

set talk off
clear
use livraria
accept 'Deseja pesquisar em qual campo?: ' to C
accept 'Procurar qual &C: ' to COND
accept 'Deseja imprimir? (S/N): ' to R
clear
if upper(R) = 'S'
    set print on
endif
go top
do while .not. eof()
if upper(&C) = upper('&COND')
    ? nome
    ? endereco
    ? trim(cidade)+'-'estado
    ? cep
    ?
    ?

```



```

    ?
endif
skip
enddo
set print off
wait 'Pressione qualquer tecla para retornar '+';
'ao menu anterior' to tecla
clear
set talk on
return

```

Programa CASO2A.PRG/CLIPPER.

```

***** Programa CASO2B.PRG/CLIPPER
***** Opera com o arquivo LIVRARIA.DBF
***** (extraido de ENDEREC.A.LBL)

```

```

use livraria
set talk off
label form endereca to print
wait 'Pressione qualquer tecla para retornar '+';
'ao menu anterior' to tecla
set talk on
return

```

Programa CASO2B.PRG/CLIPPER.

```

***** Programa CASO3.PRG/CLIPPER
***** Opera com o arquivo VENDAS.DBF

```

```

use vendas
set talk off
y = space(1)
clear
@ 10,15 say 'Escolha um numero. '+';
'Relatorio = 1. Tela = 2:' get y
read
if y = '2'
    list CLIENTE, PRODUTO, QTDE, OTN, VENC, STATUS
    wait 'Pressione qualquer tecla para retornar '+';
    'ao menu anterior' to tecla
clear
endif
if y = '1'
    report form vendas to print
    wait 'Pressione qualquer tecla para retornar '+';
    'ao menu anterior' to tecla
clear
endif
set talk on
return

```

Programa CASO3.PRG/CLIPPER.

```
***** Programa CASO4.PRG/CLIPPER
***** Atualizacao de arquivos
```

```
x = 0
do while x <> 4
clear
text
    Atualizar:
    1 = Agenda
    2 = Etiquetas
    3 = Vendas
    4 = Saida
endtext
@ 10,15 say 'Escolha um numero: ';
get x picture '9' range 1,4
read
do case
    case x = 1
        do num1
    case x = 2
        do num2
    case x = 3
        do num3
endcase
enddo
return
```

Programa CASO4.PRG/CLIPPER.

```
***** Programa NUM1.PRG/CLIPPER
***** Atualiza arquivo CLIENTES.DBF
```

```
use clientes
x = 0
do while x <> 4
clear
text
    Seleccione:
    1 = Atualiza (use PgUp PgDn . Ctrl-END para sair)
    2 = Apaga registros (por campos)
    3 = Gera: tela
        relatorio impresso
    4 = Retorna ao Menu Principal
endtext
@ 10,15 say 'Escolha um numero: ';
get x picture '9' range 1,4
read
do case
    case x = 1
        append blank
    case x = 2
        do del
    case x = 3
        do caso1
```

```

endcase
enddo
return

```

Programa NUM1.PRG/CLIPPER.

```

***** Programa NUM2.PRG/CLIPPER
***** Atualiza arquivo LIVRARIA.DBF

use livraria
x = 0
do while x < > 4
clear
text
  Seleccione:
  1 = Atualiza (PgUp PgDn e MEMO cancelados)
  2 = Apaga registros (por campos)
  3 = Gera etiquetas:
      (global = 1)
      (especifica = 2)
  4 = Retorna ao Menu Principal
endtext
@ 10,15 say 'Escolha um numero: ';
get x picture '9' range 1,4
read
do case
  case x = 1
    do tela
  case x = 2
    do del
  case x = 3
    do caso2
endcase
enddo
return

```

Programa NUM2.PRG/CLIPPER.

```

***** Programa NUM3.PRG/CLIPPER
***** Atualiza arquivo VENDAS.DBF

use vendas
x = 0
do while x < > 4
clear
text
  Seleccione:
  1 = Atualiza (PgUp PgDn. Ctrl-END para sair)
  2 = Apaga registros (por campos)
  3 = Gera: tela
      relatorio impresso
  4 = Retorna ao Menu Principal
endtext
@ 10,15 say 'Escolha um numero: ';

```

```

get x picture '9' range 1,4
read
do case
    case x = 1
        append blank
    case x = 2
        do del
    case x = 3
        do caso3
endcase
enddo
return

```

Programa NUM3.PRG/CLIPPER

```

***** Programa Del.prg/CLIPPER
***** Apaga registros

clear
accept 'Deseja pesquisar em qual campo?: 'to C
accept 'Procurar qual &C: 'to COND
accept 'Deseja ver o registro?(S/N): 'to R
clear
if upper(R) = 'S'
    list for upper(&C) = upper(COND)
endif
accept 'Quer apaga-lo?(S/N): 'to A
if upper(A) = 'S'
    clear
    delete for upper(&C) = upper(COND)
endif
list
wait '(...ENTER continua)' to tecla
x = 0
clear
@ 10,15 say 'Confirmar seu apagamento? Sim = 1. '+';
'Nao = 2.: ' get x picture '9' range 1,2
read
do case
    case x = 1
        pack
        clear
        list
        wait 'Sua tela atual. (...ENTER retorna ao Submenu)'
    case x = 2
        recall all
        list
        wait 'Sua tela atual. (...ENTER retorna ao Submenu)'
endcase
return

```

Programa Del.prg/CLIPPER.

Uma Rotina Especial da Formatação de Tela.

O Programa Tela.prg é uma rotina avançada de formatação de tela, que nos dará uma tela idêntica à tela usada no menu do *Screen Painter*, exceto por umas poucas inovações que criamos especialmente para ele. Sem esta rotina, o CLIPPER não aceitaria a característica de leitura da tela personalizada via comando APPEND BLANK, já que o CLIPPER não aceita o comando APPEND do dBASE ou acesso a telas preparadas via menu. Comentaremos especificamente este programa – e o que ele faz – no final da seção *Analisando os Procedimentos*.

***** Programa Tela.prg/CLIPPER

```

use livraria
go bottom
clear
m = space(1)
@ 1, 6 say "Banco de Dados – DISTRIBUIDORES/BRASIL"
@ 1, 55 say dtoc(date( ))
@ 4, 23 say "Consulte o menu"
@ 6, 0 say "Nome"
@ 6, 12 get LIVRARIA->NOME
@ 6, 59 say "Auxilio na Tela"
@ 8, 0 say "Endereco"
@ 8, 12 get LIVRARIA->ENDERECO
@ 8, 56 say "Cursor:  Setas"
@ 9, 56 say "Paginas:  (Proximo)"
@ 10, 0 say "Cidade"
@ 10, 12 get LIVRARIA->CIDADE
@ 10, 30 say "Estado"
@ 10, 37 get LIVRARIA->ESTADO PICTURE "AA"
@ 10, 41 say "Cep"
@ 10, 46 get LIVRARIA->CEP
@ 11, 56 say "Apagar: Esq.: Backspace"
@ 12, 64 say "Dir.: Del"
@ 13, 59 say "Sob caracter: Ctrl-G"
@ 14, 0 say "Tel"
@ 14, 12 get LIVRARIA->TEL PICTURE "(999)999-9999"
@ 15, 56 say "Memo:  (Proximo)"
@ 16, 0 get LIVRARIA->REM
@ 17, 56 say "Salvar:  Automatico"
@ 18, 56 say "Desistir:  Esc"
@ 0, 0 to 2,51 DOUBLE
@ 5, 54 to 19,79 DOUBLE
@ 7, 55 to 7,78
@ 10, 78 to 10,78
@ 10, 55 to 10,78
@ 14, 55 to 14,78
@ 16, 55 to 16,78
@ 21, 1 say "Quer mais (S/N)?: "get m
read
if LIVRARIA->NOME = ' '
    return

```

```

do num2
endif
if upper(m) = 'S'
  append blank
  replace Nome with LIVRARIA->NOME
  replace Endereco with LIVRARIA->ENDereco
  replace Cidade with LIVRARIA->CIDADE
  replace Estado with LIVRARIA->ESTADO
  replace Cep with LIVRARIA->CEP
  replace Tel with LIVRARIA->TEL
  clear
do tela
endif
return

```

Programa Tela.prg/CLIPPER.

Analizando os Procedimentos

Antes de qualquer operação de compilação, é preciso que o programador saiba preparar seus programas em dBASE, para que eles fiquem totalmente compatíveis com o compilador. E é precisamente isto o que fizemos com os 12 programas que acabamos de apresentar (ou melhor, reapresentar!).

Agora vamos comentar rápida e eficientemente cada reforma:

central.prg/CLIPPER. Note que a primeira coisa que fizemos foi acrescentar, como já havíamos previsto, “/CLIPPER” ao cabeçalho do programa para melhor identificá-lo e/ou compará-lo aos programas normais. A segunda coisa que fizemos foi diminuir o tamanho da linha @ 10,15 say ‘Escolha um número...’ . Observe que usamos para a separação da linha o apóstrofo (ou aspas) e o ponto-e-vírgula antes da instrução *get*. Lembre-se sempre, o CLIPPER não aceita linhas muito longas.

CASO1.PRG/CLIPPER. Neste programa, separamos todas as linhas consideradas longas com o apóstrofo, o sinal de mais e o ponto-e-vírgula. Aqui usamos o sinal de mais (+) porque não temos nenhuma instrução com palavra reservada (como *get*, por exemplo) e só queremos separar uma frase. Note também que usamos outro apóstrofo no reinício do comando na linha abaixo. Compare a diferença sutil entre a técnica usada em *central.prg* e *CASO1.PRG* (a propósito, usamos “*central.prg*” com letras minúsculas sem nenhuma razão especial. Você pode usar o tipo de letra que quiser).

CASO2.PRG/CLIPPER. Somente uma linha de comando foi separada com ‘+; e ’ (apóstrofo, sinal de mais, ponto-e-vírgula e, na outra linha, um apóstrofo inicial).

CASO2A.PRG/CLIPPER. Somente a instrução *Wait...* foi separada com ‘+; e ’.

CASO2B.PRG/CLIPPER. Somente a instrução *Wait...* foi separada com ‘+; e ’.

CASO3.PRГ/CLIPPER. Aqui já podemos notar (será que você pode perceber, sem verificar nos originais antigos?) uma importante modificação. Além das duas separações de linha com os habituais ‘+; e’, acrescentamos os nomes dos campos do arquivo VENDAS.DBF ao comando LIST, para que o DOS possa interpretá-lo corretamente (ou seu programa mostrará um arquivo vazio). Portanto, lembre-se de que o CLIPPER só entende um comando LIST se ele vier acompanhado dos campos que se deseja visualizar.

CASO4.PRГ/CLIPPER. Retornamos à separação antes da instrução get. Nesse caso usamos somente um apóstrofo e o ponto-e-vírgula e deixamos *get* começar a linha seguinte sem nenhuma pontuação especial.

NUM1.PRГ/CLIPPER. Para esta série de comandos que se inicia (a série NUM), temos um comentário adicional. Toda a série (NUM1, NUM2 e NUM3) será compilada, porém, somente NUM2 receberá uma tela especial para a opção 1 (Atualiza), simplesmente por motivos de estética e tamanho do livro. Entretanto, para fazer a rotina de tela personalizada para NUM1 e NUM3, simplesmente siga os mesmos passos do programa Tela.prg adiante, adaptando as instruções NOMEARQ-CAMPO aos respectivos arquivos e campos. Neste caso, ainda, todo o sistema operará corretamente, exceto a opção 1 desses dois programas. (Se você não quiser fazer as duas telas nos moldes sugeridos, outra alternativa seria adquirir o aplicativo opcional do livro que traz todos os programas prontos para sua utilização.)

Continuando com NUM1, separamos uma linha com ‘; e trocamos o comando APPEND por APPEND BLANK, já que o CLIPPER não reconhece o primeiro.

NUM2.PRГ/CLIPPER. Em NUM2 fizemos as maiores alterações, inclusive uma rotina especial chamada Tela.prg. Aqui, ajustamos o texto do menu “Selecione” e ainda efetuamos uma correção. Na seleção 1, Atualiza, avisamos que a operação das teclas PgUp e PgDn e do campo MEMO está cancelada. Isto aconteceu porque não desejamos avançar muito e terminar por fazer mais um livro dentro deste, de técnicas avançadas (Para isso, consulte sua livreria sobre **dBASE III e dBASE IV – Técnicas Avançadas**, próximas publicações desta Editora).

Note que corrigimos a opção 3, Gera etiquetas: (global = 1), (específica = 2) que antes apresentava o mesmo texto de NUM1 e NUM3.

Finalizamos com NUM2 separando uma linha com ‘; e colocando o comando DO TELA no lugar de APPEND, para que uma tela especial igual à tela do *Screen Painter* seja usada como uma rotina de comandos para que o comando APPEND BLANK possa aceitá-la.

NUM3.PRГ/CLIPPER. Alterações como em NUM1.

Del.prg/CLIPPER. Apenas uma alteração na linha @ 10,15. Mas, há algo de errado com este programa. Deixamos esse “erro” acontecer propositalmente. Tente encontrar o erro (em relação ao CLIPPER). Bem, se você não conseguiu, lá vai...

Caro leitor/usuário: note que deveríamos também ter inserido o nome dos respectivos campos com os comandos LIST, para que sua “tela atual” apareça após uma interação com este programa pelo sistema, mas preferimos deixar para você mesmo fazer isso (ou encontrar no disquete opcional). Nos programas avançados de dBASE III Plus Técnicas Avançadas (e dBASE IV Técnicas Avançadas) acrescentaremos muitas rotinas úteis e muito mais poderosas. Mas não se preocupe: os conhecimentos adquiridos até aqui serão o pré-requisito indispensável para a leitura dessas publicações. E não se esqueça de criar um programa Del para cada um, pois os nomes dos campos de cada arquivo são diferentes (a não ser que você retire o comando LIST do programa Del e a instrução WAIT seguinte (na realidade elas não são necessárias).

Tela.prg/CLIPPER. Antes de iniciar a compilação, vamos nos deter na análise mais apurada de Tela.prg. Em primeiro lugar, queremos lembrar-lhe de que esta rotina é (quase) exatamente igual ao programa TELA.SCR e TELA.FRM criado anteriormente quando aprendemos a manipular o *Screen Painter*. Porém, sem ela um programa do tipo .PRG ou compilado não pode acessar uma tela personalizada. Logicamente há outras maneiras de construir uma rotina para este serviço, mas à parte a programação de teclas especiais – omitidas objetivando o não aprofundamento do leitor num “mar sem salva-vidas” (ou num “mato sem cachorro”, se preferir) –, este programa tem tudo para otimizar seu sistema e ensinar-lhe a construir uma tela personalizada a partir do *prompt*.

No início abrimos o arquivo LIVRARIA.DBF e enviamos o *prompt* para o último registro. A seguir limpamos a tela com CLEAR e inicializamos uma variável chamada “m” (de menu), que será lida posteriormente pela última linha desse bloco de comandos, antes do comando READ. As instruções do tipo “@ say” seguem colocando, ou melhor, desenhando a tela com as coordenadas idênticas às que usamos com o menu do *Screen Painter* (você pode preparar sua tela no menu e depois usar o SIDEWAYS para gerar seu fonte. Consulte o Capítulo 15). Entretanto, logo no início inserimos duas instruções simples que não havíamos incluído na formatação com o menu. São elas, @ 1,55 say dtoc(date()), a segunda linha de comando do bloco de comandos é @ 4,23 say “Consulte o menu”, um lembrete permanente para que o usuário sempre consulte o menu que acompanha sua tela.

Na linha @ 9,56, no lugar de especificar as teclas PgUp e PgDn, indicamos com (Próximo), que somente nas próximas publicações desse autor (**dBASE III Plus – Técnicas Avançadas e dBASE IV – Técnicas Avançadas**) é que falaremos tudo a respeito de códigos de teclado, ASCII etc. O mesmo fazemos com o campo MEMO. Ele só será acessado via programas .prg numa próxima publicação. Observe também que reformamos as funções de apagamento e a função de salvamento que agora lê “Automático”.

As instruções @ TO DOUBLE são as coordenadas para desenho de linhas duplas na tela. Sem o parâmetro DOUBLE, elas desenhavam uma linha simples (@ 0,0 to 2,51 DOUBLE desenha as linhas que guardam o cabeçalho; @ 5,54 to 19,79 DOUBLE desenha a linha dupla do menu de Auxílio na Tela).

A última instrução antes de READ, pergunta se o usuário deseja continuar. Em caso posi-

tivo, finalmente o programa executa as instruções de APPEND BLANK juntamente com a rotina de instruções REPLACE, limpa a tela e retorna para mais uma sessão. Mas se não colocarmos nenhum nome no campo Nome de nossa tela personalizada, o programa então volta para o menu, o mesmo acontecendo se respondermos a “Quer mais (S/N)?” com “N”.

COMPILANDO O SISTEMA

Bem, atendidas as “formalidades”, vamos iniciar nosso processo de compilação. Primeiro saia do dBASE com QUIT. A seguir, coloque o disquete preparado de acordo com nossas especificações no início deste capítulo, ou seja, o programa CLIPPER.EXE, no acionador A. Deixe seu disquete com os programas .PRG no acionador B. Digite

CLIPPER B:CENTRAL ENTER

e, se tudo estiver de acordo, seus programas serão compilados e você receberá a seguinte tela:

```
A>clipper b:central
The Clipper Compiler, Autumn '86
Copyright (c) 1985, 1986 Nantucket Corp., All Rights Reserved.

Compiling b:CENTRAL.PRG
Compiling b:CASO1.PRG
Compiling b:CASO2.PRG
Compiling b:CASO3.PRG
Compiling b:CASO4.PRG
Compiling b:CASO2A.PRG
Compiling b:CASO2B.PRG
Compiling b:NUM1.PRG
Compiling b:NUM2.PRG
Compiling b:NUM3.PRG
Compiling b:DEL.PRG
Compiling b:TELA.PRG
Code Pass 1
Code Pass 2
Code size 3536, Symbols 576, Constants 2640
```

A>

Figura 16.1: Tela do CLIPPER com 12 programas compilados.

Neste momento, assumindo que nada de errado ocorreu, seus programas compilados serão reconhecidos apenas por um nome: CENTRAL.OBJ – o programa *objeto* a ser “linkeditado”. CENTRAL.OBJ não estará no disquete de trabalho no acionador B e sim no acionador A, portanto, transfira seu disquete com CENTRAL.OBJ (e CLIPPER.EXE) para o acionador B e coloque no acionador A o disquete contendo os programas PLINK86 e CLIPPER.LIB. A seguir acione o comando:

PLINK86 ENTER

para chamar o PLINK. Em seguida, após a mensagem de *copyright* da Phoenix Technologies Ltd., o símbolo => aparecerá na sua tela. Digite o comando:

FILE CENTRAL ENTER

que é o programa que desejamos “linkeditar”. O *prompt* => aparecerá novamente. Desta vez, digite:

OUTPUT B:CENTRAL ENTER

para fazer com que sua saída seja direcionada para o acionador B (lembre-se, no A não haverá espaço!). O nome do arquivo, neste caso, pode ser qualquer outro que preferir e não necessariamente o mesmo nome do arquivo-objeto. Para finalizar, tecla “;” (ponto-e-vírgula), para iniciar o processo de “linkagem” final. Se tudo correr bem, seu programa será “linkado” em aproximadamente 1 minuto e meio. Ao terminar, sua tela deverá estar como na Figura 16.2 – e seu sistema completamente executável no acionador B.

```
A>plink86
PLINK86plus ( Nantucket ) Version 2.21.
Copyright (C) 1984, 1985, 1986 by Phoenix Technologies Ltd.,
All Rights Reserved.
```

```
=>file central
=>output b:central
=>;
```

```
B:CENTRAL.EXE (137 K)
Insert disk with \ COMMAND.COM in drive A
and strike any key when ready
```

Figura 16.2: Tela de PLINK86 após a “linkedição”.

Esperamos que você tenha obtido sucesso na sua compilação e “linkedição”. Lembre-se de ser o mais rigoroso possível quanto à atenção dispensada para o aprendizado das técnicas simples que introduzimos aqui. Qualquer deslize certamente será fatal para a compilação e/ou “linkedição”. Lembre-se também de que muitos comandos e funções do dBASE III Plus não são suportados pelo CLIPPER (como APPEND, EDIT, BROWSE, ASSIST ou READKEY()). Recomendamos que você adquira uma publicação de bom nível sobre o assunto, para maiores esclarecimentos.

“Linkedição” via DOS

Para tornar seu programa executável via DOS, faça os mesmos procedimentos de compila-

ção anteriores, até o programa-objeto. Coloque seu disquete do MS-DOS no acionador A. Digite:

LINK ENTER

e depois retire o disquete do DOS. Coloque em seguida o disquete com o arquivo CLIPPER.LIB e o programa-objeto (CENTRAL.OBJ) no acionador A e um disquete vazio (de preferência) no acionador B.

À primeira mensagem do programa LINK, Object Modules [.OBJ]:, responda com o nome do arquivo a ser “linkeditado”, ou seja, CENTRAL. À segunda mensagem, Run File [CENTRAL.EXE]: responda com B:CENTRAL2 (ou qualquer outro nome, desde que sua saída seja direcionada para o acionador B). Você pode ignorar a terceira mensagem, se quiser, ou dar o nome de um arquivo para “mapear” seu programa. Se preferir interromper todo o processo, digite Ctrl-Break. À quarta e última mensagem, Libraries [.LIB]: digite CLIPPER. É só isso!

Após o sucesso da operação, sua tela deverá estar como na Figura 16.3.

A>LINK

**Microsoft (R) Personal Computer Linker Version 2.40
Copyright (C) Microsoft Corp 1983, 1984, 1985. All rights reserved.**

**Object Modules [.OBJ]: CENTRAL
Run File [CENTRAL.EXE]: B:CENTRAL2
List File [NUL.MAP]:
Libraries [.LIB]: CLIPPER**

A>

Figura 16.3: Tela do LINK do MS-DOS após a “linkagem”.

Capítulo

17

**LAN – Redes Locais com
dBASE III Plus**

Neste último capítulo, abordaremos o tópico *Local Area Network* de maneira conceitual e prática. Porém, nossa prática será limitada à instalação básica de um sistema em rede e aos comandos elementares para sua utilização (o assunto merece um livro exclusivo). De qualquer maneira, não se esqueça de adquirir nossa próxima publicação, **dBASE III Plus – Técnicas Avançadas**, onde mostraremos como construir sistemas inteligentes e úteis, como por exemplo, o sistema de proteção programável por senha (tal qual num sistema de redes locais) que poderá ser usado com um programa comum, entre outras dicas.

INTRODUÇÃO ÀS REDES LOCAIS

Redes Locais, neste nosso contexto, são um sistema de microcomputadores interligados que podem utilizar os mesmos recursos, *compartilhando* dados e periféricos. É um sistema econômico, quando se trata realmente de uma utilização necessária e de acesso global com ou sem *restrições*.

Entende-se por *compartilhamento de dados*, todo o potencial de rede que o sistema implantado oferece a seus usuários e por *restrições*, o acesso autorizado ou não a uma determinada informação.

As redes locais são utilizadas com frequência (ou pelo menos deviam ser, se os empresários as conhecessem melhor) em pequenas e médias empresas – sem falar nas empresas de grande porte.

Qual o Perfil Ideal do Usuário?

O tipo de usuário que deveria adquirir um sistema em rede de micros, por exemplo, é a empresa com mais de um departamento, que necessite de um controle eficaz de estoque que permita seu acesso a mais de um usuário interno, ou a vários outros departamentos. Da mesma forma, toda a contabilidade e a administração financeira também podem proporcionar acesso restrito a duas ou mais pessoas, por meio de senhas de acesso que somente usuários autorizados saberiam.

Entretanto, o mais importante é que o usuário já conheça as capacidades do dBASE, já tendo-o operado com sucesso, bem como compreendido sua capacidade de programação, o que faz deste programa, uma poderosa – e indispensável – ferramenta de programação.

As Necessidades Básicas

Para montar um sistema de rede local, é preciso ter pelo menos um computador completo, com 640K, impressora, acionador de disquete flexível e disco rígido, que será seu computador central, mais conhecido como “servidor” e um computador com pelo menos 384K, vídeo, disco rígido ou flexível e teclado, que servirá como sua estação de trabalho

ou *workstation*, como também é conhecida e que poderá ser mais de uma unidade (quanto mais unidades instaladas, maior será sua rede). Além disso você precisará de uma instalação adequada com cabos de interface etc. de acordo com as especificações locais.

O *software* utilizado pelos computadores poderá estar tanto numa estação de trabalho como no servidor, entretanto, deve-se trabalhar com o programa Administrador no servidor e o programa de Acesso na estação de trabalho. De qualquer forma, recomendamos que você procure pessoal especializado para fazer sua instalação adequadamente. Falaremos disso a seguir.

A Instalação dos Programas

Instalando o Administrador

Para fazer sua rede funcionar, primeiro você precisará de dois programas que acompanham o dBASE III Plus, o ADMINISTRATOR e o ACCESS. Verificada a existência desses dois programas crie um subdiretório com o MS-DOS (ou PC DOS)* com a instrução:

```
MKDIR B:NOMEDIR    ENTER
```

onde NOMEDIR significa o nome do subdiretório e B: o nome do acionador, se necessário.

Para transformar seu novo subdiretório no diretório *default*, use a instrução do DOS:

```
CHDIR B:\    ENTER
```

onde B: é o nome do acionador, se necessário.

Em seguida, acesse seu arquivo CONFIG.SYS para incluir nele as seguintes instruções:

```
FILES = 24  
BUFFERS = 20  
FCBS = 16,8  
LASTDRIVE = X
```

onde X representa o último drive do sistema (consulte seu manual da Ashton-Tate ou da Datalógica para maiores informações sobre o último acionador de sua rede).

Deve-se instalar o Administrador no computador servidor da seguinte forma: com o disquete do Administrador no acionador A, digite

* Consulte também o comando JOIN na versão 3.

INSTALL C:DBA ENTER

e atenda às mensagens de instalação até que sua primeira instalação esteja concluída.

Instalando o ACCESS

Para instalar o programa ACCESS, que se encontra no disquete do Sistema 1, certifique-se de que este disco não tenha sido usado (instalado). Caso já tenha instalado antes sua versão mono-usuário (aquela que usamos durante nossas interações com este livro) em seu disco rígido, use o programa (arquivo) UNINSTAL que se encontra no disquete do Sistema 1, para “desinstalar” seu dBASE. A seguir acione a instrução:

INSTALL C:ACCESS ENTER

não sem antes certificar-se do acionador designado e do subdiretório *default*.

Após a instalação ser concluída você verá uma mensagem de confirmação e o *prompt* do DOS.

Lembre-se de que o programa ACCESS deve estar numa estação de trabalho (se quiser mais instalações de ACCESS, você terá de adquirir outras cópias no seu revendedor autorizado). O ADMINISTRATOR deverá estar no computador servidor.

Tipos de Servidores de Arquivo

Além dos pacotes da Ashton-Tate, há outros servidores de arquivos populares disponíveis no mercado norte-americano. A seguir, listaremos alguns para facilitar sua pesquisa dentro de suas possibilidades de mercado. São eles:

- IBM PC Network
- (IBM Token-Ring Network)
- Novell S/Net
- (Advanced Netware V. 1.01)
- 3Com
- (3Com 3Plus Network)

Uma Dica Muito Útil

É importante que você teste todo o potencial *esperado* de sua rede local, juntamente com o pessoal especializado que terá a tarefa de usá-la. Muito cuidado com os “entendidos” que poderão lhe vender um pacote que acabará por não se adaptar à sua rede. Procure profis-

sionais competentes que possam, preferivelmente, lhe vender, instalar e testar o equipamento, bem como dar as instruções básicas para seu funcionamento.

Comandos de Inicialização

Para inicializar seu dBASE na rede que acabamos de introduzir (aqui partimos do princípio de que estamos usando a rede Token-Ring), acione o comando:

```
C:NET SHARE DFILES = C:\NOMEDIR\RWC    ENTER
```

para ativar a característica de compartilhamento do subdiretório (NOMEDIR) com as funções de leitura, gravação e criação de arquivos.

Para acessar o *prompt* do dBASE – e finalizar o processo de inicialização – a partir de uma estação de trabalho, acione o comando:

```
C:ACCESS M = A:\NOMEDIR    ENTER
```

onde C: representa o disco rígido e A: o acionador de disco flexível em uso.

Para carregar seu sistema diretamente com um determinado arquivo, você poderia digitar o comando anterior da seguinte forma:

```
C:ACCESS NOMEARQ M = A:\NOMEDIR    ENTER
```

onde NOMEARQ representa o nome de seu programa.

Lembre-se da dica sobre uma instalação competente, pois a variedade de *drives* disponíveis em sua rede pode confundir o usuário que estiver iniciando com redes locais. Isto afetará, obviamente, a sintaxe dos comandos, no que se refere à designação dos nomes (letras) do acionador em uso. Lembre-se também de atribuir o *default* ao acionador em uso.

Inicialização Via Comandos de Lote

É muito fácil efetuar a inicialização de sua rede por meio de um arquivo de comandos em lote especialmente preparado pelo sistema operacional. Após a programação do arquivo de lote, ou *batch*, se preferir, bastará acionar o nome do arquivo a partir do DOS e ele fará o resto. Vejamos então um exemplo de arquivo de lote para acionar os comandos de inicialização de nossa rede. No *prompt* do sistema operacional (A>), acione a instrução:

```
EDLIN B:MEULOTE.TXT    ENTER
```


(ou use seu processador de texto no modo não-documento para criar o pequeno arquivo que faremos a seguir).

Com a mensagem

New file

*
_

onde *New file* significa o novo arquivo (MEULOTE) que acabamos de criar e o asterisco (*) o *prompt* de EDLIN, acione o comando de inserção de EDLIN (no WordStar não será necessário este comando, mas somente a digitação do texto-programa):

I ENTER

e EDLIN exibirá sua primeira linha de comando (como nos programas em BASIC) para que você componha seu texto, ou seja, as instruções que formarão o arquivo de lote. Copie as instruções simples mostradas a seguir.

```
1: ACCESS M = A:\NOMEDIR    ENTER
2: A:ACCESS M = B:\NOMEDIR  ENTER
3: ACCESS NOMEARQ M = A:\NOMEDIR    ENTER
```

(note que isto é apenas um exemplo, por isso, não deixe de verificar os *drives* corretos a serem indicados de acordo com sua configuração).

Para retornar ao DOS e salvar todo o arquivo, tecele E (de *End*). Para usar seu arquivo MEULOTE.BAT, simplesmente digite MEULOTE no *prompt* do DOS quando você estiver preparado para o processo de inicialização do dBASE em uma rede local.

Acessando um Arquivo

Para acessar um arquivo dBASE em rede, você pode fazê-lo no modo compartilhado (o que recomendamos) ou no modo exclusivo. No modo exclusivo, não é necessário proteger o arquivo (como veremos adiante) pois ele será de uso exclusivo do usuário. Já no modo compartilhado, enquanto um departamento opera com um arquivo, outro departamento competente poderá estar atualizando este mesmo arquivo sem que um atrapalhe o outro.

O *default* para a abertura de arquivos é o modo exclusivo (exceto para arquivos do tipo imutáveis, como um arquivo .FRM, por exemplo) e para acessar o modo compartilhado, você deve acionar o comando:

SET EXCLUSIVE OFF ENTER

antes de abrir seus arquivos.

Entretanto, para evitar as eventuais “colisões” entre um acesso de um usuário e outro ao mesmo segmento de um programa, no modo compartilhado, deve-se usar as características de proteção de dados para registros ou arquivos inteiros.

Programação de Segurança

Alguns comandos disponíveis para uso em rede são fundamentais para uma programação voltada para a proteção de dados ou arquivos em uso. Já mencionamos o comando SET EXCLUSIVE. Agora falaremos sobre os comandos FLOCK, RLOCK e UNLOCK.

Usando FLOCK/UNLOCK

FLOCK pode ser usado para “trancar” um arquivo inteiro e não permitir que outro usuário acesse a mesma seção do programa que você estiver usando. Por exemplo, a pequena instrução a seguir abriria o arquivo NOMEARQ e o trancaria imediatamente:

```
use nomearq
flock( )
```

da mesma forma que a rotina

```
if flock
close database
unlock
endif
```

destrancaria o arquivo aberto.

O uso desses comandos de proteção faz com que o programa retorne valores lógicos (.T. e .F.) que mostrarão o *status* atual do programa, ou seja se estiver trancado o Administrador retornará um .F.. Como já dissemos, não pretendemos nos aprofundar no tópico sobre redes locais, portanto, recomendamos que você consulte um especialista ou um programador que esteja acostumado a trabalhar com redes, além de seu manual do usuário da Ashton-Tate ou da Datalógica, para dar continuidade a seu aprendizado sobre programação com os comandos de proteção. Não tente usar um arquivo em redes sem que as rotinas apropriadas estejam programadas, como por exemplo, uma rotina de recuperação de erro, após um número predeterminado de tentativas de acesso a um mesmo arquivo, por dois usuários ao mesmo tempo, como nosso próximo exemplo.

Usando RLOCK: Uma rotina de espera

O pequeno programa 17.1, mostra como aguardar a permissão de acesso e executar uma

rotina de recuperação de erro antes do fechamento dos arquivos

```
X = 0
DO WHILE .NOT. RLOCK( ) .AND. X<10
X = X + 1
ENDDO
IF .NOT. RLOCK( )
    DO ROTERRO
ELSE
    REPLACE...
    UNLOCK
ENDIF
```

Programa 17.1.: Rotinas para tentativa de acesso.

onde a instrução REPLACE... representa os comandos REPLACE para atualização dos registros. X representa o número de tentativas de acesso (ou seja, após 10 tentativas o programa passa à etapa seguinte).

O UTILITÁRIO PROTECT: Codificação de Segurança

O acesso a programas do dBASE III Plus em uma rede pode ter de ser autorizado, caso o usuário deseje restringir este acesso somente a pessoas autorizadas. O programa usado para gerenciar um arquivo *criptografado* (com codificação especial de segurança) chama-se DBSYSTEM.DB.

PROTECT encontra-se no disco do Administrador # 1 que já deverá estar carregado no acionador em uso. Para entrar no programa, digite

PROTECT ENTER

e você verá em seguida (após a mensagem de *copyright* característica) a tela inicial de PROTECT, como na Figura 17.1.

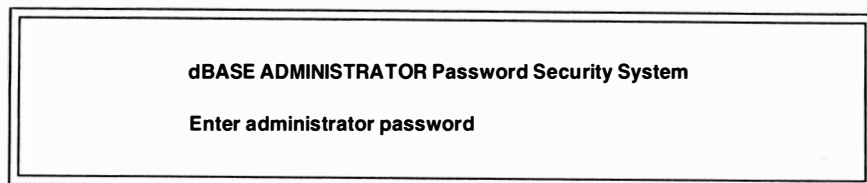


Figura 17.1: Tela de abertura de PROTECT.

Note que a tela inicial de PROTECT solicita uma senha para entrada no sistema (“*Enter administrator password*”). Você terá de digitar um nome com no máximo 8 caracteres, como na criação de um arquivo comum do dBASE. Anote a palavra num lugar seguro até que você não mais a esqueça pois o acesso à sua senha só será permitido por você mesmo!

Se este for o seu primeiro acesso, o programa solicitará a senha novamente. Confirme a mesma senha digitando-a como da primeira vez. Em seguida a tela do menu principal de PROTECT aparecerá para que você insira algumas informações. Sua tela com o menu principal deverá estar como na Figura 17.2.

Users	Files	Exit 06:35:25 pm
Login name Password Group name		
Account name Access level 2		
Store user profile		
Delete user from group		

Figura 17.2: Menu principal de PROTECT.

Nossa primeira opção é *Users*. Sob esta opção podemos ver *Login name*. Digite o nome de acesso, seu nome por exemplo, contanto que ele também não tenha mais de 8 caracteres. Em *Password*, digite sua senha (esperamos que você ainda se lembre dela). Em *Group name*, digite o nome do grupo de usuários ao qual você pertence (pode ser um departamento da empresa). Em *Account name*, pode-se acrescentar mais uma codificação para especificar uma determinada atividade. Digite o nome de sua atividade básica. Por exemplo, suponha que você trabalhe no controle de estoque de sua empresa. Assim, você poderia digitar simplesmente “Estoque”. Na opção *Access level*, especifique seu nível de acesso, que poderá ser de 1 a 8 (ou seja, do menos restrito ao mais restrito). O nível de acesso escolhido nesta fase inicial será comparado com os níveis selecionados na fase seguinte de definição dos *privilégios* de acesso, na opção *Files*. A próxima seleção, *Store user profile*, solicita que você pressione ENTER para que os dados preenchidos sejam armazenados na memória.

Para cada usuário, repita as mesmas operações. Para salvar todo o trabalho em conjunto, vá para a opção *Exit* e selecione *Save*. Para apagar um determinado usuário do grupo de usuários, simplesmente selecione a opção *Delete user from group* e salve tudo de novo com *Exit* e *Save*. Sua tela deverá se parecer com a tela da Figura 17.3.

Users	Files	Exit 06:45:28 pm
-------	-------	------------------

Login name	GOMES
Password	TINTAS
Group name	GRUPO
Account name	Estoque
Access level	2
Store user profile	
Delete user from group	

Figura 17.3: Menu personalizado do usuário.

Acessando um Arquivo Privilegiado

Após suas interações com a primeira opção, *Users*, vá para a opção *Files* e selecione, pressionando ENTER, a opção *Select new file*. Observe que um menu aparecerá à direita de sua tela atual, contendo todos os arquivos de seu sistema. Novamente pressione ENTER sobre a opção desejada para selecionar o arquivo com o qual trabalharemos daqui por diante. A seguir, sob as regras de privilégios especiais de acesso a arquivos, selecione o grupo de usuários (um dos que já foram inseridos no sistema na tela de *Users*) junto à opção *Group name*. Em seguida, selecione o nível do privilégio de leitura desse usuário. Lembramos que o nível mais restrito é o nível 8 e o menos restrito, o nível 1. Tecle o nível escolhido na opção *Read privilege level*. Faça o mesmo nas opções *Update privilege level*, para atualizar seu nível de privilégio, *Extend privilege level*, para ampliar esse nível, e *Delete privilege level*, para apagar o nível de privilégio.

Agora, sob o comando dos privilégios de acesso aos campos do arquivo selecionado anteriormente, tecle o número desejado na opção *Establish access level* e na opção *Establish field privileges* (esta opção proporciona 3 tipos de privilégio que se alternam à medida que pressionamos ENTER. São elas: FULL – privilégios total –, R/O – privilégio somente para leitura – e NONE – não é permitida leitura ou gravação. O controle dessas opções pode ser efetuado por meio do menu lateral que exibe os campos e suas condições no momento da seleção). Em seguida, armazene tudo pressionando ENTER sobre a opção *Store file privileges*. Se quiser cancelar as entradas dessa tela, tecle ENTER sobre a última opção, *Cancel current entry*. Sua tela de *Files* poderá estar como na Figura 17.4.

Repita os procedimentos que acabamos de executar, para preencher novas telas para grupos de usuários diferentes, até que todos os grupos tenham sido inseridos. Para salvar todo o trabalho, isto é, todas as telas preenchidas, vá para *Exit* e selecione a opção *Save*. (Caso você prefira abandonar todo o trabalho, selecione a opção *Abandon* ao invés de *Save*.)

Users

Files

Exit 06:56:18 pm

Select new file	
File access privileges	
Group name	
Read privilege level	7
Update privilege level	7
Extend privilege level	7
Delete privilege level	7
Filed access privileges	
Establish access level	2
Establish field privileges	R/O
Store file privileges	
Cancel current entry	

Figura 17.4: Menu da opção Files de PROTECT.

O Arquivo Criptografado

Uma vez dentro de PROTECT, todos os seus arquivos tornam-se criptografados, isto é, protegidos contra acesso indevido por meio de uma codificação especial fornecida pelo próprio programa. Para cada arquivo criptografado, permanece uma cópia intacta do programa original. Os arquivos criptografados recebem automaticamente a extensão .CRP. Para maior segurança, remova os arquivos originais com extensão .DBF (certifique-se primeiro de ter uma cópia criptografada) e renomeie seus arquivos .CRP para extensões do tipo .DBF quando for necessário.

As Opções de Saída e Salvamento

Para sair do programa e armazenar todo o trabalho, use a opção **Exit**. *Save* salva todo o trabalho e permanece no PROTECT. *Abandon* cancela tudo o que foi feito e ainda permanece no PROTECT. Para retornar para o sistema operacional e salvar todo o trabalho, use a opção **Exit**.

Após a seleção de saída via **Exit**, o arquivo especial DBSYSTEM.DB armazena as senhas, os níveis de acesso e as demais informações de cada usuário e todos os bancos de dados em uso são criptografados. Qualquer tentativa de uso do dBASE III Plus via ACCESS será seguida de uma tela especial de identificação do usuário. Caso sua identificação não esteja correta, não haverá como acessar o sistema!

COMANDOS E FUNÇÕES DO ADMINISTRADOR

Na Tabela 17.1, fornecemos os comandos adicionais específicos do Administrador para uso programável num ambiente de rede local (alguns já vistos neste capítulo):

Comandos de Rede	
DISPLAY STATUS	Exibe o <i>status</i> do sistema e sua condição especial de proteção.
LIST STATUS	Idêntico a DISPLAY STATUS, porém exibe as informações diretamente, sem pausa.
DISPLAY USERS	Exibe os nomes dos usuários ligados ao sistema.
LIST USERS	Idêntico a DISPLAY USERS, porém exibe as informações diretamente, sem pausa.
SET ENCRYPTION	O <i>default</i> é ON, o que significa proteção automática. OFF desativa a proteção automática.
SET EXCLUSIVE	O <i>default</i> é ON. Em OFF permite-se o modo de compartilhamento de dados sem “colisões”.
SET PRINTER	Comando de redirecionamento da saída impressa. Um exemplo de sintaxe é: SET PRINTER TO LIPT1.
TO PRINT	Esta opção pode ser acrescentada aos comandos DISPLAY/LIST para impressão imediata.
 <i>Funções</i>	
FLOCK()	Protege (“tranca”) um arquivo.
RLOCK()	Protege (“tranca”) um registro.
UNLOCK	Desprotege (“destranca”) um arquivo/registro na área atual de trabalho. UNLOCK ALL exerce a mesma função em todas as áreas de trabalho.

Tabela 17.1: Comandos de Rede.

Bem, esperamos que você tenha ficado satisfeito com nosso trabalho. Se isto aconteceu, aguarde nossas próximas publicações sobre dBASE III Plus e dBASE IV – o mais recente lançamento da Ashton-Tate.

Consulte sua livreria de preferência, ou o distribuidor nacional mais próximo, na relação que incluímos no Apêndice D.

Software Opcional

Consulte o Apêndice D sobre como adquirir e utilizar o disquete opcional contendo todos os programas utilizados prontos para uso em seu PC.

O Apêndice A, a seguir, relaciona, por ordem alfabética, todos os comandos do dBASE III Plus, bem como suas funções, convenções e operadores.

Obrigado!

Apêndice

A

Índice de Comandos e Funções do dBASE III Plus

Neste apêndice relacionamos uma lista dos comandos e funções do dBASE III Plus – inclusive os comandos não utilizados neste livro. Os comandos que forem compatíveis com o compilador CLIPPER aparecerão marcados como [CLIPPER] e os comandos que não são aceitos por este compilador não serão marcados, como APPEND, por exemplo. Todos os comandos que forem compatíveis com a instrução TO PRINT (gera uma cópia impressa) serão marcados com [TO PRINT]. Comandos cuja descrição não inclui a sintaxe (em negrito) significam que sua sintaxe é somente a digitação do próprio comando.

COMANDOS	DESCRIÇÃO/SINTAXE
A->	Identificam um campo de arquivo aberto na área de trabalho 1 ou A, selecionada via comando SELECT. JOIN WITH VENDAS TO NOVOARQ FOR A-> CLIENTE = B-> CLIENTE FIELDS CLIENTE, PRODUTO, QTDE
ACCEPT	Armazena um <i>string</i> em uma variável de memória. [CLIPPER] accept 'Deseja pesquisar em qual campo?: ' to C
ALIAS	Permite acesso ao arquivo com troca de nomes. USE CLIENTES ALIAS PACIENTES
ALL	Como complemento de um comando, usado para identificar todos os registros do arquivo em uso. DELETE ALL FOR SNOPE = 'Silva'
APPEND	Acrescenta um registro no arquivo .DBF.
APPEND BLANK	O mesmo que APPEND, porém acessa diretamente o final do arquivo. [CLIPPER]
APPEND FROM	Acrescenta um registro de outro arquivo ao final do arquivo atual. [CLIPPER] APPEND FROM NOMEARQ
ASSIST	Acessa o menu. ASSIST ou F2
AVERAGE	Calcula a média aritmética. [CLIPPER] AVERAGE QTDE FOR CLIENTE = '001'
B:	Identifica este tipo de acionador de disquete. SET DEFAULT TO B
B->	Identifica um campo de arquivo aberto na área de trabalho 2 ou B, selecionada via comando SELECT. JOIN WITH VENDAS TO NOVOARQ FOR B-> CLIENTE = A-> CLIENTE FIELDS CLIENTE, PRODUTO, QTDE

BROWSE	Edição com tela panorâmica.
/C	Usado com o comando SORT para ignorar letras maiúsculas ou minúsculas em uma operação de classificação. SORT ON NOME/C, SNAME/C TO NOVONOME
CALL	Executa arquivo de memória carregado com LOAD. [CLIPPER]
CANCEL	Cancela a execução do programa. [CLIPPER]
CHANGE	Edita campos especificados de um arquivo. Pode ser aplicado na edição de campos MEMO. em programas do tipo .PRG. CHANGE FIELD CEP FOR CITY = 'SPaulo'
CLEAR	Limpa a tela. [CLIPPER]
CLEAR ALL	Fecha todõs os bancos de dados e arquivos e limpa todas as variáveis de memória. [CLIPPER]
CLEAR FIELDS	Zera todos os campos inicializados com SET FIELDS. [CLIPPER]
CLEAR GETS	Zera variáveis GET para a próxima instrução READ. [CLIPPER]
CLEAR MEMORY	Apaga as variáveis de memória. [CLIPPER]
CLEAR TYPEAHEAD	Zera o <i>buffer</i> do teclado.
CLOSE	Fecha arquivos especificados. [CLIPPER] CLOSE DATABASE
CONTINUE	Continua a busca de um registro especificado por LOCATE. [CLIPPER]
COPY FILE	Copia qualquer tipo de arquivo. [CLIPPER] COPY FILE TELA.FMT TO TELA.TXT
COPY STRUCTURE	Copia a estrutura de um banco de dados criando, simultaneamente, um outro banco de dados. COPY STRUCTURE EXTENDED copia um registro e um cabeçalho juntamente com a estrutura. [CLIPPER] COPY STRUCTURE TO NOVOARQ

COPY TO	Copia o arquivo em uso para outro arquivo. COPY TO NOMEARQ FIELDS NOME, SNAME
COUNT	Calcula o número de registros que atendem a uma determinada condição. [CLIPPER] COUNT FOR CLIENTE = '001' TO NOMEARQ
CREATE	Cria um novo arquivo a partir de outro criado via comando COPY STRUCTURE EXTENDED. [CLIPPER] CREATE NOMEARQ FROM EXTARQ
CREATE LABEL	Cria um arquivo de etiquetas de endereçamento. CREATE LABEL NOMEARQ
CREATE QUERY	Cria um arquivo-relatório especial. CREATE QUERY NOMEARQ
CREATE REPORT	Cria um arquivo para definição do relatório. CREATE REPORT NOMEARQ
CREATE SCREEN	Cria um menu especial para formatação de tela personalizada. CREATE SCREEN NOMEARQ
CREATE VIEW	Cria uma relação de inter-ligação entre bancos de dados múltiplos. CREATE VIEW NOMEARQ
/D	Usado com o comando SORT para classificações em ordem decrescente. SORT ON NOME/D TO NOVONOME
DELETE	Marca asteriscos para posterior apagamento. [CLIPPER] DELETE RECORD 10
DELIMITED	Copia arquivos .DBF para outros formatos de arquivo. COPY TO CARTA.TXT DELIMITED
DIR	Exibe os arquivos de um diretório. [CLIPPER]
DISPLAY	Exibe registros e campos do arquivo atual. [CLIPPER]
DISPLAY HISTORY	Exibe os comandos previamente executados.
DISPLAY MEMORY	Exibe as variáveis de memória atuais.
DISPLAY STATUS	Exibe informações sobre o estado atual dos programas ativos.

DISPLAY STRUCTURE	Exibe a estrutura atual do banco de dados.
DO	Executa um programa do tipo .PRG. [CLIPPER] DO CENTRAL
DO CASE	Executa uma das condições especificadas. [CLIPPER]
DO WHILE	Executa um <i>loop</i> no programa até que uma determinada condição seja atendida. [CLIPPER] DO WHILE .NOT. EOF()
EDIT	Edita o conteúdo de um arquivo.
EJECT	Executa um avanço de formulário. [CLIPPER]
ENDTEXT	Fecha uma instrução TEXT. [CLIPPER]
ERASE	Apaga o arquivo especificado. [CLIPPER] ERASE NO-MEARQ
EXIT	Sai de um <i>loop</i> DO WHILE antes do término da execução do programa. [CLIPPER]
EXPORT TO	Cria arquivos para o programa PFS <i>file</i> . EXPORT TO NOMEARQ
FIND	Encontra o registro especificado num arquivo indexado. [CLIPPER] FIND 'Silva'
GET	Usado com o comando READ para aceitar uma variável definida pelo usuário. [CLIPPER] @ 10, 15 Say 'Escolha um numero: 'GET X
GO BOTTOM	Vai para o último registro de um arquivo. [CLIPPER]
GO TO	Vai para o registro especificado. [CLIPPER] GO TO RECORD 70
GO TOP	Vai para o primeiro registro de um arquivo. [CLIPPER]
HELP	Auxílio na tela. HELP ou F1
IF	Execução condicional de um comando. [CLIPPER] if y = '2' use clientes endif

IMPORT FROM	Cria um arquivo a partir do programa PFS <i>file</i> . IMPORT FROM NOMEARQ
INDEX ON	Cria um arquivo indexado por um campo de um arquivo em uso. [CLIPPER] INDEX ON NOME TO NOVONOME
INPUT	Aguarda pela entrada de uma resposta (numérica) do usuário. [CLIPPER] INPUT 'Quantas vezes?' TO NUMERO
INSERT BEFORE	Insere um novo registro numa determinada posição. GO TO 10 INSERT BEFORE
JOIN TO	Combina registros e campos de dois arquivos, formando um terceiro arquivo. [CLIPPER] JOIN TO NOVOARQ FOR CLIENTE = B-> CLIENTE
JOIN WITH	Combina registros e campos de dois arquivos, formando um terceiro arquivo. [CLIPPER] JOIN WITH NOMEARQ TO NOVOARQ FOR CLIENTE = NOMEARQ -> CLIENTE FIELDS
LABEL FORM	Gera etiquetas de endereçamento via arquivo criado com MODIFY LABEL . [TO PRINT] [CLIPPER] LABEL FORM ETIQUETA TO PRINT
LIST	Exibe o arquivo atual na tela. [TO PRINT] [CLIPPER]
LIST FOR	Exibe campos especificados pelo usuário, no arquivo atual. [TO PRINT] LIST FOR CIDADE = 'RJ' TO PRINT
LIST HISTORY	Exibe os últimos comandos executados. [TO PRINT]
LIST STATUS	Exibe o estado atual do programa. [TO PRINT]
LIST STRUCTURE	Exibe a estrutura atual do arquivo. [TO PRINT]
LOAD	Coloca um arquivo <i>assembly</i> na memória para posterior execução via CALL .
LOCATE	Localiza um registro especificado. [CLIPPER] LOCATE FOR SNAME = 'Silva'
LOOP	Salta todos os comandos entre ele mesmo e um <i>loop</i> DO WHILE . [CLIPPER]

M->	Especifica uma variável de memória. É usado especialmente para designar uma variável que tenha o mesmo nome de um campo em uso. ? M-> CLIENTE
MODIFY COMMAND	Edita/cria um programa .PRG. MODIFY COMMAND NO-MEARQ
MODIFY LABEL	Edita/cria um programa .LBL. MODIFY LABEL NOMEARQ
MODIFY QUERY	Edita/cria um programa .QRY. MODIFY QUERY NO-MEARQ
MODIFY REPORT	Edita/cria um programa .FRM. MODIFY REPORT NO-MEARQ
MODIFY SCREEN	Edita/cria um programa .SCR. MODIFY SCREEN NO-MEARQ
MODIFY STRUCTURE	Modifica a estrutura de um banco de dados.
MODIFY VIEW	Edita/cria um programa .VUE. MODIFY VIEW NOMEARQ
NOTE/*	Indica linha de comentário. [CLIPPER]
ON ERROR	Executa um comando especificado quando da ocorrência de um erro. ON ERROR DO NOMEARQ
PACK	Apaga registros marcados com DELETE. [CLIPPER]
PARAMETERS	Especifica variáveis de memória de DO WHILE. [CLIPPER] PARAMETERS <i>Relação de parâmetros</i>
PRIVATE	Especifica variáveis de memória que serão apagadas após o término do programa. [CLIPPER] PRIVATE NOMEVAR
PROCEDURE	Identifica o início de um procedimento num programa .PRG. [CLIPPER] PROCEDURE NOMEPROC
PUBLIC	Especifica variáveis de memória que não são apagadas quando do término do programa. [CLIPPER] PUBLIC NOMEVAR
QUIT	Abandona o dBASE e retorna ao sistema operacional. [CLIPPER]

RANGE	Especifica uma faixa de entrada de valores aceitáveis pelo programa. [CLIPPER] @ 10, 15 say 'Escolha um numero: 'get X picture '9' range 1,5
READ	Complementa os comandos @ SAY e GET para permitir a entrada de uma resposta ao usuário. [CLIPPER] @ 10,15 say 'Escolha um numero: 'get X picture '9' RANGE 1,5 read
RECALL	Cancela a marca de apagamento de um ou mais registros. [CLIPPER] RECALL RECORD 10
REINDEX	Recria os índices de um arquivo indexado. [CLIPPER]
RELEASE	Apaga variáveis de memória especificadas. Não apaga variáveis públicas, ou incluídas em arquivos de comando. [CLIPPER] RELEASE ALL
RENAME	Muda o nome de um arquivo. [CLIPPER] RENAME NO-MEARQ.DBF NOVOARQ.DBF
REPLACE	Troca o conteúdo de campos especificados. [CLIPPER] Replace Nome with LIVRARIA -> NOME
REPORT FORM	Cria um relatório usando um formato previamente criado do arquivo atual. [TO PRINT] [CLIPPER] Report form vendas to print
RESTORE FROM	Recupera variáveis de memória salvas. [CLIPPER] RESTORE FROM NOMEARQ
RESUME	Continua um programa suspenso.
RETRY	Repete a tentativa de execução. Usado em rede.
RETURN	Retorna ao programa chamador ou ao menu principal. [CLIPPER]
RUN	Roda um programa externo. [CLIPPER] RUN TIME ou RUN 123 (Para rodar o Lotus 1-2-3)
SAVE TO	Salva a memória atual em um arquivo de memória. [CLIPPER] SAVE TO NOMEARQ
SAY	Usado com @ para posicionar uma saída na tela. [CLIPPER] @ 10, 15 say 'Escolha um numero:'

SEEK	Posiciona o cursor no primeiro registro de um arquivo indexado quando o valor de uma tecla iguala a expressão especificada. [CLIPPER] SEEK CHR(13)
SELECT	Seleciona arquivos em uso para uma das dez áreas de trabalho numeradas de 1 a 10, ou de A a J. [CLIPPER] SELECT A USE CLIENTES INDEX CLIENTES SELECT B USE VENDAS INDEX VENDAS SELECT A
SET	Exibe o menu de comandos SET.
SET ALTERNATE ON/OFF	Armazena o que tiver sido bloqueado com SET ALTERNATE TO <i>nome do arquivo</i> . [CLIPPER]
SET ALTERNATE TO	Define um arquivo especial para armazenamento no formato .TXT. [CLIPPER] SET ALTERNATE TO NOMEARQ
SET BELL ON/OFF	Ativa/desativa o sinal sonoro. [CLIPPER]
SET CARRY ON/OFF	Traz para o próximo registro vazio, o último registro de uma tela APPEND.
SET CATALOG ON/OFF	Ativa/desativa um catálogo de arquivos, definido pelo comando SET CATALOG TO <i>nome do arquivo</i> .
SET CATALOG TO	Define um arquivo-catálogo (um arquivo do tipo diretório, que, por sua vez, conterá todos os arquivos que forem guardados pelo usuário no catálogo). SET CATALOG TO NOMEARQ
SET CENTURY ON/OFF	Ativa/desativa a exibição do século atual em uma data.
SET COLOR ON/OFF	Ativa/desativa os modos de cores e monocromático. [CLIPPER]
SET COLOR TO	Ajusta os atributos de cores de um monitor colorido – e, com algumas limitações –, também de um monitor monocromático. [CLIPPER] SET COLOR TO B/W
SET CONFIRM ON/OFF	Avanço do cursor, sem necessidade de se pressionar ENTER, após o preenchimento de um campo. [CLIPPER]

SET CONSOLE ON/OFF	Ativa/desativa a visualização de dados na tela. [CLIPPER]
SET DATE	Ativa os formatos de data definidos pelo usuário. São eles: AMERICAN, ANSI, BRITISH, FRENCH, GERMAN, ITALIAN. (No formato ANSI o ano é o primeiro; no formato GERMAN o dia vem primeiro; no formato AMERICAN o mês aparece primeiro.) SET DATE AMERICAN
SET DEBUG ON/OFF	Direciona a saída do comando SET ECHO ON para a impressora, para fins de rastreamento de um programa.
SET DECIMALS TO	Especifica o número de casas decimais de uma operação de cálculo. [CLIPPER] SET DECIMALS TO 2
SET DEFAULT TO	Especifica o acionador (<i>drive</i>) a ser usado para trabalho. [CLIPPER] SET DEFAULT TO B
SET DELETED ON/OFF	Ignora, quando em OFF (não exibe) registros marcados para deleção com o asterisco. [CLIPPER]
SET DELIMITER ON/OFF	Determina o modo de entrada dos campos exibidos, como vídeo inverso, etc... [CLIPPER]
SET DELIMITER TO	Especifica delimitadores de campo para exibição na tela. [CLIPPER] SET DELIMITER TO “[]”
SET DEVICE TO	Determina a saída de comandos formatados. [CLIPPER] SET DEVICE TO PRINTER
SET DOHISTORY ON/OFF	Armazena um “histórico” dos comandos já utilizados pelo usuário.
SET ECHO ON/OFF	Determina se os comandos executados por um programa serão exibidos para rastreamento.
SET ESCAPE ON/OFF	Ativa/desativa a tecla Esc.
SET EXACT ON/OFF	Determina como os <i>strings</i> serão comparados: se EXACT estiver ON, então o conteúdo dos campos fornecidos pelo usuário deverá conter o tipo exato de apresentação do conteúdo dos campos do programa. [CLIPPER]
SET FIELDS ON/OFF	Ativa/desativa uma lista de campos especificados pelo comando SET FIELDS TO.

SET FIELDS TO	Determina o acesso aos campos por comandos diretos. SET FIELDS TO ALL
SET FILTER TO	Exibe somente os registros que atendam a uma condição especificada. Esta condição pode ser somente um determinado conteúdo de um campo. [CLIPPER] SET FILTER TO "Aguias" \$ENDERECO
SET FIXED ON/OFF	Ativa/desativa o comando SET DECIMALS. [CLIPPER]
SET FORMAT TO	Seleciona um arquivo de formatação. [CLIPPER] SET FORMAT TO NOMEARQ
SET FUNCTION TO	Programa as teclas de função. [CLIPPER] SET FUNCTION 10 TO "USE VENDAS"; APPEND;
SET HEADING ON/OFF	Ativa/desativa a geração de cabeçalhos e/ou títulos.
SET HELP ON/OFF	Ativa/desativa a mensagem <i>Do you want some help?</i> (Y/N) que ocorre quando o dBASE percebe que você cometeu algum engano.
SET HISTORY ON/OFF	Ativa/desativa um "histórico" de comandos. O <i>default</i> é de 20 comandos, mas pode-se alterá-lo para mais.
SET HISTORY TO	Determina o número de linhas a ser armazenado em um "histórico" de linhas. SET HISTORY TO 50
SET INDEX TO	Seleciona e abre um arquivo indexado especificado. [CLIPPER] SET INDEX TO NOMEARQ
SET INTENSITY ON/OFF	Ativa/desativa o vídeo inverso. [CLIPPER]
SET MARGIN TO	Controla a margem esquerda da impressora. Seu <i>default</i> é zero. [CLIPPER] SET MARGIN TO 5
SET MEMOWIDTH TO	Especifica o tamanho da coluna de um campo memo. SET MEMOWIDTH TO 80
SET MENU ON/OFF	Ativa/desativa a exibição dos menus.
SET MESSAGE TO	Determina uma mensagem, definida pelo usuário, na parte inferior da tela. [CLIPPER] SET MESSAGE TO "Pressione ESC para sair"

SET ORDER TO	Cria um arquivo indexado sem precisar fechar ou abrir arquivos. SET ORDER TO NOME
SET PATH TO	Seleciona uma via de acesso a um arquivo. [CLIPPER] SET PATH TO B:\NOMEDIR
SET PRINT ON/OFF	Envia a saída diretamente para a impressora (ON) ou somente para a tela (OFF). [CLIPPER]
SET PRINTER TO	Redireciona a impressão para o dispositivo especificado pelo DOS. SET PRINTER TO LPT1
SET PROCEDURE TO	Seleciona (cria) um arquivo de procedures. [CLIPPER] SET PROCEDURE TO NOMEARQ
SET RELATION TO	Conecta dois bancos de dados por um campo em comum. [CLIPPER] SET RELATION TO NOME INTO NOMEARQ
SET SAFETY ON/OFF	Ativa/desativa as mensagens de alerta para a sobregravação de arquivos.
SET SCOREBOARD ON/OFF	Ativa/desativa a exibição de informações da linha de <i>status</i> .
SET STATUS ON/OFF	Ativa/desativa a exibição da barra de <i>status</i> na parte inferior da tela.
SET STEP ON/OFF	Ativa/desativa o rastreamento de um programa linha-a-linha.
SET TALK ON/OFF	Ativa/desativa a exibição de resultados de cálculos e comandos na tela (ativa/desativa a exibição dos números de registro).
SET TITLE ON/OFF	Ativa/desativa o <i>prompt</i> para a descrição de um arquivo se o catálogo estiver ativo.
SET TYPEAHEAD TO	Controla o número de caracteres no <i>buffer</i> do teclado. O <i>default</i> é 20. SET TYPEAHEAD TO 50.
SET UNIQUE ON/OFF	Ativa/desativa a impressão de registros não duplicados. (Usado com INDEX). [CLIPPER]
SET VIEW TO	Abre um arquivo do tipo .VUE. SET VIEW TO NOMEARQ

SKIP	Move o ponteiro do registro atual para frente ou para trás, conforme indicação. [CLIPPER] SKIP -1
SORT TO	Cria um arquivo com registros classificados pela ordem especificada. [CLIPPER] SORT ON NOME TO NOVONOME
STORE	Armazena um valor ou uma expressão em uma variável de memória. [CLIPPER] STORE 1 TO X
SUM	Efetua soma. [CLIPPER] SUM QTDE
SUSPEND	Interrompe a execução de um programa.
TEXT	Exibe um bloco de texto em um programa. É fechado com ENDTEXT. [CLIPPER]
TOTAL ON	Cria, resume e totaliza um arquivo para outro arquivo. [CLIPPER] TOTAL ON OTN TO NOMEARQ
TYPE	Exibe ou imprime um arquivo. [TO PRINT] [CLIPPER] TYPE CENTRAL.PRG TO PRINT
UPDATE ON	Atualiza o arquivo, substituindo dados de outro arquivo. [CLIPPER] UPDATE ON CLIENTE FROM VENDAS REPLACE QTDE WITH B-> QTDE
USE	Abre um arquivo especificado na área atual de trabalho. [CLIPPER] USE VENDAS
WAIT	Interrompe a execução de um programa até que uma tela seja pressionada. [CLIPPER] wait 'Pressione qualquer tecla para retornar ao menu principal' to tecla
ZAP	Apaga permanentemente um arquivo. [CLIPPER]

ÍNDICE DE FUNÇÕES POR CATEGORIA

Caracteres

AT()	Pesquisa de <i>substring</i> [CLIPPER]
FUNCTION()	Ver Conversões.
LEFT()	Seleção para a esquerda.

LOWER()	Converte em minúscula. [CLIPPER]
LTRIM()	Remove espaços em branco anteriores. [CLIPPER]
PICTURE()	<i>Ver</i> Conversões.
REPLICATE()	Repete a expressão. [CLIPPER]
RIGHT()	Seleção para a direita.
RTRIM()	Remove espaços em branco posteriores.
SPACE()	Cria espaços.
STUFF()	Insere parte de um <i>string</i>
SUBSTR()	Cria <i>substring</i> [CLIPPER]
TRANSFORM()	Caracteres/números no formato PICTURE. [CLIPPER]
TRIM()	Remove espaços em branco entre dois campos. [CLIPPER]
UPPER()	Converte para maiúsculas. [CLIPPER]

Conversões

ASC()	Converte para código ASCII. [CLIPPER]
CHR()	Converte de ASCII para caractere. [CLIPPER]
FUNCTION()	Usado com GET para atribuir modelos de entrada de dados pré-estabelecidos.
PICTURE()	Usado com GET para atribuir modelos de entrada de dados pré-estabelecidos. [CLIPPER]
STR()	Converte número para caractere. [CLIPPER]
VAL()	Converte caractere para número. [CLIPPER]

Data e Hora

CLOW()	Converte data no dia da semana. [CLIPPER]
--------	---

CMONTH()	Converte data no mês. [CLIPPER]
CTOD()	Converte caractere em data. [CLIPPER]
DATE()	Data do sistema operacional. [CLIPPER]
DAY()	Dia do mês (numérico). [CLIPPER]
DOW()	Dia da semana (numérico – a partir da variável do usuário). [CLIPPER]
DTOC()	Conversão data para caractere. [CLIPPER]
MONTH()	Mês atual (numérico – a partir da variável do usuário) [CLIPPER]
TIME()	Hora do sistema operacional. [CLIPPER]
YEAR()	Ano (numérico – a partir da variável do usuário). [CLIPPER]

Entrada de Dados via Teclado

INKEY()	Rastreamento do teclado em busca de uma determinada tecla pressionada pelo programador. [CLIPPER]
READKEY()	Saída da tela atual para atender a um comando determinado pela próxima linha de comando, ao pressionamento da tecla especificada pelo programador.

Gerais

BOF()	Início do arquivo. [CLIPPER]
COL()	Coluna atual da tela. [CLIPPER]
DELETED()	Registro marcado para deleção. [CLIPPER]
DISKSPACE()	Espaço livre do disco atual.
EOF()	Fim do arquivo. [CLIPPER]
ERROR()	Número da condição ON ERROR.

FILE()	Verifica a existência de um arquivo. [CLIPPER]
FOUND()	Verdadeiro quando encontra o registro especificado. [CLIPPER]
IIF()	Variação de IF. Uma ou outra condição.
ISALPHA()	Verifica a existência de uma letra.
ISCOLOR()	Verifica o uso de monitor a cores.
ISLOWER()	Verifica a ocorrência de uma letra minúscula.
ISUPPER()	Verifica a ocorrência de uma letra maiúscula.
LEN()	Comprimento de um <i>string</i> . [CLIPPER]
LUPDATE()	Última atualização do arquivo. [CLIPPER]
MESSAGE()	Mensagem de ON ERROR.
PCOL()	Posição de coluna na impressora. [CLIPPER]
PROW()	Posição de linha na impressora. [CLIPPER]
RECCOUNT()	Número de registros no arquivo. [CLIPPER]
RECNO()	Número do registro atual. [CLIPPER]
RECSIZE()	Tamanho do registro.
ROW()	Linha atual da tela. [CLIPPER]
TYPE()	Testa uma expressão. [CLIPPER]

Identificação

DBF()	Nome do banco de dados em uso.
FIELD()	Nome dos campos do banco de dados.
FKLABEL()	Nome das teclas de função.
FKMAX()	Número máximo de teclas de função.

GETENV()	Obtém variáveis do DOS.
NDX()	Nome dos arquivos indexados atuais.
OS()	Nome do sistema operacional.
VERSION()	Número da versão atual do dBASE.

Matemática

ABS()	Valor absoluto.
EXP()	Exponenciação.
INT()	Inteiro.
LOG()	Logaritmo.
MAX()	Maior de dois valores.
MIN()	Menor de dois valores.
MOD()	Resto.
ROUND()	Arredondamento. [CLIPPER]
SQRT()	Raiz quadrada. [CLIPPER]

Símbolos e Sinais

!	Converte para maiúsculas. [CLIPPER]
?	Exibe o comando na próxima linha. [CLIPPER]
??	Exibe o comando na linha atual. [CLIPPER]
;	Separa uma linha de comando longa. [CLIPPER]
^	Exponenciação. [CLIPPER]
***	(um ou mais asteriscos). Usado como início de linha de comentário em programas do tipo .PRG. [CLIPPER]

**	Exponenciação. [CLIPPER]
*	Multiplicação. [CLIPPER]
+	Soma. [CLIPPER]
-	Subtração. [CLIPPER]
/	Divisão. [CLIPPER]
()	Usado com macros e substituições. [CLIPPER]
<	Menor que. [CLIPPER]
<=	Menor que ou igual. [CLIPPER]
>	Maior que. [CLIPPER]
> =	Maior que ou igual. [CLIPPER]
=	Igual. [CLIPPER]
#	Diferente de. [CLIPPER]
< >	Diferente de. [CLIPPER]
&	Macro. [CLIPPER]
&&	Usado como linha de comentário em complemento a uma linha de comando. [CLIPPER]
\$	Usado para marcar um determinado caractere a ser pesquisado dentro de um campo.
@	Usado para inicializar instruções de resposta do usuário, ou para desenho em tela. [CLIPPER]

No Apêndice B, apresentamos as principais extensões de nomes de arquivo do dBASE III PLUS, do DOS e dos principais programas do mercado, entre outras.

Apêndice

B

Extensões de Nomes de Arquivo

Este apêndice pretende ser uma simples abordagem às principais extensões de nomes de arquivo relacionadas aos principais programas e/ou tarefas executadas por um arquivo, disponíveis no mercado internacional.

EXTENSÕES DO dBASE III Plus

Extensão	Descrição
.CAT	Arquivos-catálogo (diretório de arquivos).
.DBF	Arquivos de banco de dados.
.DBT	Arquivos <i>Memo</i> .
.EXE	Arquivos executáveis.
.FMT	Arquivos de tela personalizada.
.FRM	Arquivos de relatório.
.HLP	Arquivos de auxílio em tela.
.LBL	Arquivos de etiquetas de endereçamento.
.LIB	Arquivos-biblioteca.
.MEM	Arquivos de memória temporária.
.NDX	Arquivos indexados.
.OBJ	Arquivos-objeto.
.OVL	Arquivos de <i>Overlays</i> .
.PRG	Arquivos de programação.
.QRY	Arquivos <i>Query</i> .
.SCR	Arquivos de formatação de tela.
.TXT	Arquivos-texto.
.VUE	Arquivos <i>View</i>

Arquivos Especiais

Extensão	Descrição
.BAK	Arquivos-reserva (<i>Backup</i>).
.BAT	Arquivos de lote.
.CRP	Arquivos criptografados (Redes locais).
.DB	(CONFIG.DB): Arquivos de configuração.
.DB	(DBSYSTEM.DB): Arquivo especial de codificação de rede.
.DBB	Arquivos (.DBF) de conversão dBASEII-dBASE III Plus.
.PRB	Arquivos (.PRG) de conversão dBASE II-dBASE III Plus.

EXTENSÕES GERAIS – DOS E OUTROS

Extensão	Descrição
.ASM	Arquivos <i>Assembly</i> .
.BAS	Arquivos de programa em BASIC.
.BLD	Arquivos binários (quando carregado via BLOAD, do BASIC).
.BIN	Arquivos binários.
.CBL	Arquivos de código-fonte de programa COBOL.
.COD	Arquivos de código-objeto compilado.
.COM	Programa executável, armazenado exatamente igual ao programa armazenado na memória. (Arquivos .EXE são realocados para uma posição específica na memória após o carregamento).
.CRF	Arquivos-fonte de referência cruzada.
.DAT	Arquivos de dados.
.DIF	Arquivos de intercambiamento de dados. <i>Data Interchange File</i> . Arquivos ASCII de conversão de programas de planilhas de cálculo.

.FMT	Arquivos de formatação usado por processadores de texto.
.FOR	Arquivos de código-fonte de programas FORTRAN.
.LIB	Arquivos de biblioteca usados por compiladores.
.LST	Arquivos de listagens de programas destinadas à impressão.
.HEX	Arquivos hexadecimais (Comando DEBUG do DOS).
.MAP	Arquivos de “mapeamento” de “linkagem” de programas.
.REF	Arquivos de listagem de referência cruzada.
.REL	Arquivos <i>Assembly</i> realocáveis.
.SYS	Arquivo de configuração de sistema para carregamento de programas.
.TMP	Arquivo temporário.
.\$\$\$	Arquivo temporário (especialmente os arquivos com defeito de gravação).

EXTENSÕES DOS PRINCIPAIS PROGRAMAS

Extensão	Descrição
.FW	Arquivos Framework.
.PAS	Arquivos Pascal
.PRN	Arquivos SuperCalc.
.TXT	Arquivos-texto em geral.
.VC	Arquivos VisiCalc.
.WRK	Arquivos Symphony.
.WK1	Arquivos Symphony V.1.1 (etc).
.WKS	Arquivos Lotus 1-2-3.
.PFS	Arquivos PFS <i>file</i> .

EXTENSÕES DE INTERFACE ENTRE ARQUIVOS

Extensão	Descrição
.DIF	Planilhas de cálculo.
.PFS	PFS <i>file</i> .
.PRN	SuperCalc, Symphony.
.SDF	Processadores de texto, Multimate.
.WKS	Lotus 1-2-3.

Códigos de Conversões e Interfaces

A seguir relacionamos os principais códigos de interface entre programas.

Extensão	Descrição
ASCII DELIMITED SDF	Processadores de texto.
DELIMITED	Bancos de dados.
PRN SYLK	Planilhas de cálculo, gráficos.
WKS	Lotus 1-2-3.

No Apêndice C, apresentamos a estrutura completa dos menus do dBASE III PLUS.

Apêndice

C

Estrutura Completa dos Menus

Set Up

Database file

Select a database file.
(Seleciona um arquivo.)

Format for Screen

Select a screen design for updating with APPEND and EDIT.
(Seleciona um formato de tela para atualização com APPEND e EDIT.)

Query

Select a query file to filter records in this database file.
(Seleciona um arquivo *Query* para condições específicas de um registro.)

Catalog

Select a catalog from which to choose files.
(Seleciona um arquivo em um diretório.)

View

Select a view to use.
(Abre arquivo *View*.)

Quit dBASE III PLUS

Finish this session of ASSIST and QUIT dBASE III PLUS
(Termina a sessão e sai do programa.)

Create

Database file

Create a database file structure.
(Cria um banco de dados.)

Format

Create a screen design.
(Cria um formato de tela.)

View

Create a view file.
(Cria um arquivo *View*.)

Query

Create a query to access specified records in this database file.
(Cria um arquivo *Query* para acesso a registros especificados.)

*Report***Create a report layout.**

(Cria um formato de relatório.)

*Label***Create a label format.**

(Cria um formato de etiquetas de endereçamento.)

*Update**Append***Add new records to the bottom of this database file.**

(Acrescenta novos registros ao final deste arquivo.)

*Edit***Edit the contents of this database file, one record at a time.**

(Edita o conteúdo deste arquivo, um registro por vez.)

*Display***Display the contents of this database file.**

(Exibe o conteúdo deste arquivo.)

*Browse***Edit the contents of this database file, several records at a time.**

(Edita o conteúdo deste arquivo, vários registros por vez.)

*Replace***Update the individual fields of this database file.**

(Atualiza campos individuais deste arquivo.)

*Delete***Mark specified records for deletion.**

(Marca registros especificados para deleção.)

*Recall***Restore specified records marked for deletion.**

(Restaura registros marcados para deleção.)

*Pack***Permanently erase records marked for deletion.**

(Apaga permanentemente os registros marcados para deleção.)

Position

Seek

Search for an index key that matches the specified expression.

(Busca uma chave de indexação que atenda a uma expressão especificada.)

Locate

Search for the first record meeting a specified condition.

(Busca o primeiro registro que atenda à condição especificada.)

Continue

Search for next record meeting the LOCATE condition.

(Busca o próximo registro que atenda à condição de LOCATE.)

Skip

Position the file pointer by skipping records.

(Salta registros de acordo com a especificação.)

Goto Record

Move the file pointer to a specific record number.

(Move-se para o registro especificado.)

Retrieve

List

List the contents of this database file to the screen or printer.

(Lista o conteúdo deste arquivo na tela ou impressora.)

Display

Display the contents of this database file.

(Exibe o conteúdo deste arquivo na tela.)

Report

Display this database file using an existing report layout.

(Exibe este arquivo usando um formato de relatório existente.)

Label

Display this database file using an existing label format.

(Exibe este arquivo usando um formato de etiquetas existente.)

Sum

Display the totals of the specified numeric fields.

(Exibe o total dos campos numéricos especificados.)

Average

Display the average of specified numeric fields.

(Exibe a média dos campos numéricos especificados.)

Count

Display the record count of this database file.

(Exibe a contagem dos registros deste arquivo.)

*Organize**Index*

Create an index file to access records in a specified order.

(Cria um arquivo indexado para acessar registros na ordem especificada.)

Sort

Create a sorted database file.

(Cria um arquivo classificado.)

Copy

Copy records from one database file to another.

(Copia registros de um arquivo para outro.)

*Modify**Database file*

Change the structure of this database file.

(Muda a estrutura deste arquivo.)

Format

Change the structure of an existing screen design.

(Muda a estrutura de um formato de tela existente.)

View

Change the structure of an existing view file.

(Muda a estrutura de um arquivo *View* existente.)

Query

Change the structure of an existing query file.

(Muda a estrutura de um arquivo *Query* existente.)

Report

Change the structure of an existing report layout.

(Muda a estrutura de um formato de relatório existente.)

Label

Change the structure of an existing label format.

(Muda a estrutura de um formato de etiquetas existente.)

Tools

Set drive

Change the default disk drive.

(Muda o acionador de disquete *default*.)

Copy file

Duplicate the contents of an existing file to create a new file.

(Duplica o conteúdo de um arquivo existente criando um novo arquivo.)

Directory

Display the contents of a disk directory.

(Exibe o conteúdo do diretório de um disco.)

Rename

Rename an existing file.

(Renomeia um arquivo existente.)

Erase

Remove an existing file from the directory.

(Apaga um arquivo existente do diretório.)

List structure

Display the structure of this database file.

(Exibe a estrutura deste arquivo.)

Import

Import a PFS file into dBASE III PLUS.

(Insere um arquivo PFS no dBASE III PLUS.)

Export

Export a dBASE III PLUS file to a PFS file.

(Envia um arquivo dBASE III PLUS para um arquivo PFS.)

No Apêndice D apresentamos um guia de serviços e informações úteis, incluindo informações sobre a utilização do Disquete Opcional que contém todos os programas deste livro.

Apêndice

D

Guia de Serviços e Informações Úteis

Este apêndice visa a levar ao usuário/programador, as informações básicas sobre os principais serviços disponíveis no mercado nacional relacionados com este livro.

DISQUETE OPCIONAL

Não encontrando na livraria mais próxima, ou na livraria de sua preferência, o **Disquete Opcional** contendo todos os programas do livro – prontos para uso com seu dBASE III Plus em seu PC – pode ser adquirido em nossos distribuidores nacionais e/ou revendedores relacionados abaixo (por ordem alfabética de estado):

Preço: 4 OTNs

Bahia

Dmerval da Costa Chaves Cia. Ltda.
R. Padre Vieira, 9 – Salvador – BA
Cep.: 40020 – Tel.: (071) 245-1235

Livraria Grandes Autores Ltda.
Av. Antonio Carlos Magalhães, 260 – Salvador – BA
Cep.: 41830 – Tel.: (071) 248-2248

Ceará

Sérgio Braga Barbosa
R. Dom Joaquim, 284 – Aldeota – Fortaleza – CE
Cep.: 61110 – Tel.: (085) 221-3111

Distrito Federal

Livraria Técnica Ltda.
CLS – 102 Bloco A – Lj. 17 – Brasília – DF
Cep.: 70000 – Tel.: (061) 321-4338

Maranhão

Oliveira Costa
R. Joaquim Távora, 353 – São Luiz – MA
Cep.: 65000 – Tel.: (098) 222-0107

Minas Gerais

Livraria Importadora Científica Ltda.
Av. Augusto de Lima, 233 – S/lj. 13/17 – Belo Horizonte – MG
Cep.: 30190 – Tel.: (031) 224-5791

Editora Itatiaia Ltda.

R. São Geraldo, 67 – Belo Horizonte – MG

Cep.: 30150 – Tel.: (031) 222-8630

Pernambuco

Livro 7 Empreendimentos Culturais Ltda.

R. 7 de Setembro, 329 – Recife – PE

Cep.: 51000 – Tel.: (081) 231-5213

Paraná

Guerreiro & Guerreiro Ltda.

R. Treze de Maio, 218 – lj. 2 – Curitiba – PA

Cep.: 80510 – Tel.: (041) 234-7430

Rio de Janeiro

CITEC – Ciência e Tec. Livr. Edit. Ltda.

R. São Januário, 907 – lj. A – Rio de Janeiro – RJ

Cep.: 20921 – Tel.: (021) 580-5480

Ciência Moderna Computação Ltda.

Av. Rio Branco, 156 – Sub-solo 127 – Rio de Janeiro – RJ

Cep.: 20043 – Tel.: (021) 262-5723

Rio Grande do Sul

SAGRA – Liv. Edit. Distribuidora Ltda.

R. João Alfredo, 448 – Porto Alegre – RS

Cep.: 90050 – Tel.: (0512) 27-5222

Santa Catarina

Editora e Livraria Lunardelli Ltda.

Av. Vitor Meirelles, 28 – Florianópolis – SC

Cep.: 88010 – Tel.: (0482) 23-0689

São Paulo

DISAL – Distr. Associados de Livros Ltda.

R. Vitória, 486/496 – São Paulo – SP

Cep.: 01210 – Tel. (011) 221-1011

LITEC – Livraria Editora Técnica Ltda.
R. dos Timbiras, 257 – São Paulo – SP
Cep.: 01208 – Tel.: (011) 222-0477

EBRAS – Editora Brasileira Ltda.
R. Borges Lagoa, 1044 – São Paulo – SP
Cep.: 04038 – Tel.: (011) 549-8633

Livraria Cultura Editora Ltda.
Av. Paulista, 2073 Lj. 153 – São Paulo – SP
Cep.: 01311 – Tel.: (011) 285-4033

Instruções Para Uso do Disquete Opcional

O Disquete Opcional de dBASE III Plus – Guia Completo, de Roger Dalton, contém todos os programas do livro, em sua versão completa, ou seja, arquivos que são manipulados durante as sessões de programação do livro e que, no final das sessões, apresentarem-se incompletos (como exemplos de apagamentos de arquivos etc.), permanecem com a versão considerada completa.

Todos os programas foram incluídos – dos mais simples aos mais complexos – bem como suas variações, como por exemplo, a série de programas que compõem os sistemas .PRG (os SuperMacros), que encontram-se no Disquete Opcional completos sob seus verdadeiros nomes (central.prg, AUTOPROC.PRG, CENTRAL.EXE).

Carregue normalmente seu sistema operacional MS-DOS ou compatível e a seguir seu dBASE III Plus – *System # 1* e *System # 2*, de acordo com os procedimentos já aprendidos, ou com seu manual original da Datalógica/Ashton-Tate.

Para saber os tipos de arquivos residentes em seu Disquete Opcional, certifique-se de estar no modo de operação do menu *Assist* (caso esteja no *Prompt*, tecle F2). Em seguida vá para a opção *Tools* e selecione *Directory* (pressionando ENTER) e *B*: para receber um menu de tipos de arquivo, como a Figura AD.1.

.dbf	Database Files
.ndx	Index Files
.fmt	Format Files
.lbl	Label Files
.frm	Report Files
.txt	Text Files
.vue	View Files
.qry	Query Files
.scr	Screen Files
.*	All Files

Figura AD.1: Submenu de Directory.

Selecione a opção desejada e você verá todos os arquivos existentes do tipo escolhido. (Se preferir, selecione ** All files* e todos os arquivos de seu disquete serão exibidos.)

Agora, já sabendo como acessar o diretório de seus disquetes de trabalho, para usá-los eficientemente, primeiro abra um arquivo *.DBF* qualquer para que seu acesso relativo aos menus lhes seja garantido. Vá para a opção *Set Up* e selecione *Database file, B:*, *NO-MEARQ, N* – onde *NOMEARQ* significa o nome do arquivo escolhido.

Consulte o capítulo correspondente ao tipo de arquivo com o qual você deseja trabalhar. Como o capítulo obviamente mostrará não só como operar com seu programa, mas também como “construí-lo”, consulte a parte que mais lhe interessar, de acordo com seu próprio nível de conhecimento do dBASE III Plus.

Qualquer dúvida, telefone ou escreva para a Ebras no endereço fornecido neste livro ou, em caso de defeito (muito pouco provável), você poderá trocar seu disquete por outro igual junto ao seu revendedor. Bom proveito!

PRINCIPAIS DISTRIBUIDORES DE SOFTWARE

A seguir relacionamos os distribuidores nacionais dos softwares mencionados neste livro (por ordem alfabética de *software*).

Clipper

Officer Computadores Ltda.
Av. Pompéia, 1811 – Cep. 05023. São Paulo – SP.
Tel. (011) 262-4755. Telex (011)80479.

dBASE II, dBASE III, dBASE III Plus e Framework

Datalógica Sistemas e Comércio de Software Ltda.
Av. Paulista, 1842 – 9º andar – Cep. 01310. São Paulo – SP.
Tel. (011) 283-0355. Telex (011) 32645.

Lotus 1-2-3, Sideways, Symphony

Intercorp do Brasil Ltda.
Rua Lauro Muller 116 – Gr. 3302 – Cep. 22290. Rio de Janeiro – RJ.
Tel. (021) 541-9449. Telex (021) 35969.

MS-DOS, SuperCalc, SideKick, Turbo Basic

Compucenter Informática Ltda.

Rua Bela Cintra, 987 – 10º andar – Cep. 01415. São Paulo – SP.

Tel. (011) 257-0577. Telex (011) 21689.

WordStar

Brasoft Produtos de Informática Ltda.

Av. Angélica 1996 – Cep. 01228. São Paulo – SP.

Tel. (011) 231-2555. Telex (011) 34579.

EQUIPAMENTOS UTILIZADOS

Foram utilizados os seguintes equipamentos na produção deste livro. (Endereços no final):

Micro

Todos os programas deste livro foram adaptados e testados num microcomputador XTEC – PC Super turbo, de 704K, da XT Tecnologia Ltda., adquirido na Ciência Moderna Computação, incluindo teclado Digiponto e vídeo Angra, pelo autor – nos originais – e por profissionais da área.

O conjunto demonstrou bom equilíbrio com destaque para o excelente serviço de manutenção da XT Tecnologia quando da fase de instalação do micro, o profissionalismo da equipe de vendas da Ciência Moderna Computação, que não poupou esforços para a realização dos testes elementares, a excelente resolução do vídeo Angra e o desenho ergométrico do teclado Digiponto.

Impressora

Foi utilizada para a fase de impressão dos testes deste livro, uma impressora Grafix – GS1000 da Scritta Eletrônica Ltda., adquirida na Ciência Moderna Computação, tendo sido atendidas todas as necessidades do Editor, até o presente momento.

Diagramação, Composição e Montagem

Os originais do Autor/Editor foram entregues em formulário contínuo impresso para a RioTexto Tecnologia e Processamento Ltda., onde foram diagramados, compostos e arte-finalizados por processo eletrônico com especificações do Editor.

Endereços dos Fornecedores

Ciência Moderna Computação Ltda.

Av. Rio Branco, 156 – Sobreloja 217 – Cep. 20043. Rio de Janeiro – RJ.

Tel. (021)262-3906.

RioTexto Tecnologia e Processamento Ltda.

Rua do Catete, 311 Gr. 816 – Flamengo – Cep. 22230. Rio de Janeiro – RJ.

Tel. (021)285-2915.

XT Tecnologia Ltda.

Praça Prof. Pinheiro Guimarães, 24 – Tijuca – Cep. 20530. Rio de Janeiro – RJ.

Tel.(021) 278-4097.

Índice Analítico

Aplicativo

- Disquete opcional, 348
- Distribuidores, 351
- Instruções para uso, 350
- Distribuidores nacionais, 165

Arquivos

- .DBF, 24
- Apagamento, 61, 97
- Armazenamento, 34
- Capacidade, 91
- Classificação, 56, 66
- Classificações múltiplas, 74
- Controle do cursor, 33
- Copiando:
 - no dBASE, 56
 - no MS-DOS, 12, 15
- Criação, 24
- Editoração, 174
- Estruturação, 28
- Extensões de nomes, 336
- Fechando, 91
- Imprimindo
 - Ver Relatórios
- Indexando, 143
- Label, 193
- Listando, 40
- Nome, 24
- Operações com datas, 126
- Preenchimento, 26
- Query, 102
- Recuperação, 64
- Report, 180
- Separando, 143
- Unindo, 147, 148, 151
- View, 197
- Visualizando, 85, 89

Banco de dados

- Definição, 3
- Estrutura, 2
- Menu, 28
- Modificação, 82
- Múltiplo, 142

Campos

- Contando, 125
- Do Screen Painter, 239
- Eliminando, 84
- Especificando, 45
- Inserindo, 83
- Listando, 41
- Memo, 176, 250
- Somando, 115
- Totalizando, 136
- Trocando, 82

Clipper

- Ver Software

dBASE II

- Ver Software

dBASE III Plus

- Auxílio na tela, 21
- Capacidade, 27
- Carregando o sistema, 18
 - Disco rígido, 19
 - Discos do sistema, 18
 - Sistema em rede, 20
- Comandos
 - Como abreviar, 222
 - Índice geral, 318
 - Memória dos comandos do teclado, 219
 - SET, 223
- Entrando no programa, 18
- Funções,
 - Índice geral, 318
- Saindo do programa, 20
- Teclas de função, 52
 - Ver Menu principal
 - Ver Menus
- Disquete opcional
 - Ver Aplicativo
- Distribuidores nacionais
 - Ver Aplicativo

IBM PC

- Teclado, 25
- Teclas de controle de cursor, 25
- Vídeo
 - Atributos de cores, 221

Impressão

- Ver Arquivos
- Ver Programação

Lotus 1-2-3

- Ver Software

Menu Principal

- Entrando, 18
- Linha de status, 28
- Movimentação na tela, 20
- Procedimento de uso, 38
- Saindo, 20
- Tela, 19

Menus

- Create, 24
- De Programação, 163
- Estrutura completa, 342
- Modify, 82
- Organize, 56
- Position, 50
- Principal, 19
- Retrieve, 42
- Screen Painter, 236
- Set Up, 19
- Tools, 97

- Update, 30
- MS-DOS
 - Ver Sistema operacional
- PFS File
 - Ver Software
- Programação
 - Etiquetas de endereçamento, 164, 195
 - Imprimindo, 280
 - Impressora, 276
 - Introdução, 138
 - Instalação em redes locais, 305
 - Macros, 164
 - Menu, 163
 - Procedures, 228
 - Programando
 - CASO1.PRG, 208
 - CASO2.PRG, 209
 - CASO2A.PRG, 210
 - CASO2B.PRG, 211
 - CASO3.PRG, 211
 - CASO4.PRG, 212
 - Del@.PRG, 215
 - Menu Central, 205
 - NUM1.PRG, 213
 - NUM2.PRG, 214
 - NUM3.PRG, 214
 - Tela.PRG, 296
 - Programas .PRG, 158
 - Projeto e desenvolvimento, 204
 - Técnicas de depuração, 216, 221
 - Teclas de função, 52
 - Telas personalizadas, 236, 254
- Redes locais
 - Acesso a um arquivo, 308
 - Privilegiado, 312
 - Arquivo criptografado, 313
 - Comandos de inicialização, 307
 - Comandos e funções do Administrador, 314
 - Instalação, 305
 - Necessidades básicas, 304
 - Perfil ideal do usuário, 304
 - Programação de segurança, 309
 - Tipos de servidores de arquivo, 306
 - Utilitário Protect, 310
- Registros
 - Apagando, 61
 - Busca rápida, 77
 - Contagem, 125
 - Continuação de busca, 51
 - Especificados, 49
 - Listando, 63
 - Soma, 115
- Relatórios
 - .FRM, 184
 - .LBL, 194
 - .PRG, 199
 - .TXT, 260
 - .VUE, 197
- SideKick
 - Ver Software
- Sideways
 - Ver Software
- Sistema operacional
 - Comandos, 6
 - COPY, 15
 - DIR, 9
 - DIR/W, 10
 - DISKCOPY, 12
 - ERASE, 16
 - FORMAT, 11
 - FORMAT/V, 11
 - RENAME, 14
 - Data, 127
 - Hora, 136
 - Inicialização, 6
- Software
 - Clipper, 288
 - dBASE II, 284
 - Interface com dBASE III Plus, 261
 - LINK do DOS, 302
 - Lotus 1-2-3, 270
 - MS-DOS
 - Ver Sistema operacional
 - PFS File, 284
 - SideKick, 264
 - Sideways, 261
 - Turbo Basic, 268
 - Ver dBASE III Plus
 - Ver Distribuidores
 - WordStar, 276
- Turbo Basic
 - Ver Software
- Variáveis de memória
 - Arredondamento, 123
 - Divisão, 119
 - Exponenciação, 124
 - Logaritmo, 124
 - Média, 124
 - Multiplicação, 119
 - Potenciação, 124
 - Privadas, 137
 - Públicas, 137
 - Soma, 118
- WordStar
 - Ver Software

Seleções da Biblioteca EBRAS

LOTUS 1.2.3 Aplicações Financeiras

Robert Williams

Preparado especialmente para todos os usuários do *software* Lotus 1.2.3 em qualquer aplicação que envolva cálculos financeiros. É, portanto, ferramenta indispensável ao bom desempenho de quem possui o programa e precisa obter resultados precisos e imediatos. Com 10 capítulos/programas, Lotus 1.2.3 Aplicações Financeiras apresenta de forma didática, dezenas de ilustrações que compõem as mais variadas simulações em um PC.

200 p. (Ver Disquete 1)

LOTUS 1.2.3 Técnicas Avançadas

David C. Rier e Edmund S. Fine

Primeiro livro de Lotus avançado no Brasil, Lotus 1.2.3 Técnicas Avançadas começa onde outras publicações sobre o 1.2.3 terminam – com as características avançadas que o notabilizaram como o programa mais procurado do momento. Inclui novos conceitos de programação com Macros, Criação de Gráficos quase perfeitos, Classificação em Banco de Dados, Comunicações (micro-mainframe, BASIC-1.2.3 e vice-versa) etc..

283 p. (Ver Disquete 2)

LOTUS 1.2.3 Macros

Nelson Alves

Primeira apresentação do Lotus 1-2-3 Versão 2.01 no Brasil, Lotus 1-2-3 Macros inclui tópicos como a Linguagem de Comando de Macros – para versões 1.A, 2.0 e Symphony, diversos macros aplicativos para as áreas financeiras, Macro para Mala Direta – incluindo Geração de Etiquetas de Endereçamento, Macro de transferência de Arquivos entre o Lotus 1-2-3 e o Mainframe, dBASE II/III, VisiCalc, Arquivos DIF, Jazz, Lotus 1.A etc. Inclui documentação completa de todos os programas.

143 p. (Ver Disquete 3)

Controle Automático de Processos

Robert A. Mollenkamp

Este livro foi elaborado para demonstrar os Métodos Fundamentais de Engenharia de Desenvolvimento, Projeto e Aplicação de Sistemas de Controle de Processos em Nível Básico. É dada maior ênfase à aplicação prática do que a um tratamento teórico carregado com cálculos matemáticos. O texto examina as características dos processos industriais e suas influências na escolha da estratégia de controle. É dirigido para engenheiros, técnicos e outras pessoas envolvidas com o Projeto e Operação de Sistemas de Controle de Processos que possuem pouca ou nenhuma experiência, beneficiando também os gerentes e o pessoal de vendas.

169 p.

MS-DOS - Incluindo Guia Completo de Referência

Chris DeVoney

Abordagem prática e completa sobre o sistema operacional da Microsoft. Este manual procura abranger, de forma didática, todo o potencial do MS-DOS e deve ser considerado *leitura indispensável* para o usuário que pretende interagir com o PC com versatilidade e precisão. MS-DOS Incluindo Guia Completo de Referência deve ser lido como material de suporte ao Manual da Microsoft, ou separadamente como texto didático.

294 p.

Sistema Operacional UNIX - Um Guia Conceitual

James R. Groff/Paul N. Weinberg

Descrito na PC MAGAZINE como "o tipo de livro que é geralmente vendido para grandes empresas por centenas de dólares a cópia", Sistema Operacional UNIX - Um Guia Conceitual inclui uma comparação entre o UNIX e outros sistemas operacionais como CP/M e MS-DOS e suas outras versões incluindo XENIX, Berkeley, System V, além de uma visão de suas perspectivas no futuro. Excelente investimento para todo tipo de usuário interessado em uma análise técnica e conceitual do sistema operacional UNIX. *Inclui guia completo de referência.*

193 p.

dBASE III Guia Completo

Alan Simpson

dBASE III Guia Completo traz para o leitor a abordagem mais técnica e prática entre todas as publicações do mercado. Com dezenas de programas e aplicações, dBASE III Guia Completo inclui Métodos de Interfaceamento com BASIC, dBASE II, WordStar e SuperCalc; Técnicas Avançadas, Dicas Práticas, Projeto de Desenvolvimento de Programas, Introdução aos Bancos de Dados e um Guia Completo de Referência.

260 p. (Ver **Disquete 4**)

WordStar – CP/M – MS-DOS

Arthur Naiman

"Best Seller" da Sybex Computer Books, este livro apresenta a mais completa publicação deste fabuloso processador de palavras disponível no mercado. Completo, incluindo Acentuação em Português e Guia de Referência, WordStar – CP/M – MS-DOS pode ser considerado o padrão de livro para os usuários – e futuros usuários – deste *software*.

240 p. (Ver **Disquete 5**)

CP/M Incluindo Introdução aos Micros, CP/M-86 e MP/M

Nelson Alves

Provavelmente o mais completo manual sobre CP/M do mercado, este livro abrange todos os comandos e funções do CP/M nas duas versões 2.2 e 3.0, além do farto material de Introdução aos Microcomputadores (que pode ser ignorado caso o usuário seja avançado) e o Guia de Referência do CP/M, CP/M-86 e MP/M. Excelente leitura para todos os usuários deste Sistema Operacional - avançados ou iniciantes.

160 p.

Apple - Melhores Programas

Nelson Alves

Apple - Melhores Programas (incluindo BASIC avançado) apresenta uma coletânea dos 28 "melhores programas" para aplicação imediata no computador APPLE e compatível. Dividido em 2 partes de interesse, este livro sugere uma abordagem prática e muitas vezes avançada da linguagem BASIC em Jogos, Resolução de Problemas Matemáticos, Programas Práticos, Pessoais e Educacionais.

140p. (Ver **Disquete 6**)

VisiCalc Aplicações Comerciais

*Robert E. Williams
Bruce J. Taylor*

VisiCalc Aplicações Comerciais é uma coletânea de programas voltados para o profissional/empresário e para todos aqueles que gerenciam atividades administrativas e financeiras, inclusive pessoais. Com 12 programas/simulações, VisiCalc Aplicações Comerciais apresenta, de forma didática, grande número de ilustrações que poderão auxiliar o usuário do VisiCalc nas mais variadas aplicações.

190 p. (Ver **Disquete 7**)

SuperCalc Curso Completo – Incluindo SuperCalc²

*Mitchell Waite/Sharyn Venit
Diane Burns*

Indicado para estudantes e professores da área de microinformática – com 9 lições que completam todo o ciclo de operação de uma planilha eletrônica padrão – incluindo três apêndices especiais abrangendo: Referência de Funções Especiais e Matemáticas, Referência dos Comandos de Barra (apresentando os menus originais) e Documentação das Planilhas.

254 p.

MSX - Melhores Programas

Nelson Alves

MSX – Melhores Programas (incluindo BASIC-MSX avançado) apresenta uma coletânea dos 22 “melhores programas” para aplicação imediata no computador MSX e compatível. Dividido em três áreas de interesse, este livro sugere uma abordagem prática e muitas vezes avançada do BASIC-MSX em Jogos, Resolução de Problemas Matemáticos, Programas Práticos, Pessoais e Educacionais.

135 p. (Ver Cassete 1)

RS-232 Técnicas de Interface

Joe Campbell

RS-232 Técnicas de Interface é leitura indispensável para os usuários de microcomputadores em geral, que desejarem um acesso mais aprofundado – e prático – ao dispositivo de interface mais “compatibilizado” e conhecido do momento. Abrangendo protocolos de Handshaking, Modems, Estudo de Casos, Comparações Técnicas, entre outros tópicos, este livro, primeiro sobre o assunto no Brasil, vem preencher esta importante lacuna no mercado nacional – suporte técnico a todos os níveis!

160 p.

Desenvolvimento Estruturado de Sistemas

Carlos Gomes/José Carlos Santos Jorge/Pimenta Filho

É a primeira publicação inteiramente nacional sobre a aplicação da linguagem de programação COBOL Estruturada em um Sistema de Controle de Vendas (SCV) real. Indicado para estudantes e professores de Processamento de Dados, programadores/estudantes de COBOL, analistas de sistemas e profissionais da área e, como suporte essencial a professores e estudantes de Administração de Empresas, Economia e Contabilidade nas matérias de Informática. Inclui Metodologia de Desenvolvimento Estruturado, Programação Estruturada e Técnicas Avançadas de COBOL – VSAM, Balance Line, Comando Call, Manipulação de Tabelas etc.

278 p.

MARKETING

Matemática Financeira Com Aplicações em Microcomputadores & Planilha de Cálculo

Juan Carlos Lapponi

Livro didático e abrangente, para profissionais e iniciantes em Matemática Financeira e/ou Planilhas Eletrônicas de Cálculo – 16 bits. Orientado para a Análise Quantitativa dos assuntos financeiros em geral. O leitor desenvolve gradualmente os conhecimentos dos princípios financeiros, sua sequência, equação de problemas e, simultaneamente, a aplicação destes conhecimentos em Planilhas de Cálculo.

260 p. (Ver Disquete 8)

Assembly IBM PC Técnica de Programação

Alan R. Miller

“Best Seller” da Sybex Computer Books, este livro traz tudo o que um programador precisa para trabalhar em um ambiente IBM PC. Tópicos específicos, nunca antes abordados em livro, são incluídos: as interrupções do BIOS e do ROM BIOS, o programa DEBUG de depuração do sistema, expansão da memória por software, conversão WordStar – ASCII, proteção de arquivos, reprogramação de teclado, endereçamento de portas, as funções de interrupção 21 hex do DOS, e muito mais.

325 p. (Ver Disquete 9)

Marketing Racional – Do Produto ao Consumidor

Antonio P. Pinheiro

A Teoria Racional, defendida pelo professor Pinheiro, é a grande estratégia deste trabalho. Pela primeira vez aborda-se, com ênfase idêntica, os dois lados interessados na questão – o Produtor e o Consumidor.

Dirigido a estudantes e profissionais de Marketing, este livro é inteiramente voltado para o problema mercadológico brasileiro e deve ser leitura obrigatória a todos os interessados no tema.

140 p.

DISQUETE OPCIONAL

1. LOTUS 1.2.3 Aplicações Financeiras. *Software* disponível na versão plastificada com o selo "Incluindo Disquete" ou separadamente. Contém os 10 programas do livro, com menu personalizado, prontos para utilização em um PC.

2. LOTUS 1.2.3 Técnicas Avançadas. *Software* disponível na versão plastificada com o selo "Incluindo Disquete" ou separadamente. Contém todas as aplicações do livro, com menu personalizado, prontas para utilização em um PC.

3. LOTUS 1.2.3 Macros. *Software* disponível (versão 2.01) na versão plastificada com o selo "Incluindo Disquete" ou separadamente. Contém todos os programas do livro prontos para utilização em seu PC.

4. dBASE III Guia Completo. *Software* disponível na versão plastificada com o selo "Incluindo Disquete" ou separadamente. Contém todas as aplicações do livro, prontas para utilização em um PC.

5. WordStar – CP/M – MS-DOS. *Software* disponível na versão plastificada com o selo "Incluindo Disquete" ou separadamente. Contém 40 modelos de cartas Comerciais, Internacionais, Etiquetas de Endereçamento, Textos Legais e Literários em geral.

6. Apple - Melhores Programas. *Software* disponível na versão plastificada com o selo "Incluindo Disquete" ou separadamente. Contém os 28 programas do livro prontos para utilização em seu Apple II.

7. VisiCalc Aplicações Comerciais. *Software* disponível na versão plastificada com o selo "Incluindo Disquete" ou separadamente. Contém os 12 programas do livro, com menu personalizado, prontos para utilização em um PC.

8. Matemática Financeira com aplicação em Microcomputador & Planilha de Cálculo. *Software* disponível na versão plastificada com selo "Incluindo Disquete" ou separadamente. Contém todas as aplicações do livro, prontas para utilização em seu PC.

9. Assembly IBM PC Técnicas de Programação. *Software* disponível na versão plastificada com o selo "Incluindo Disquete" ou separadamente. Contém os 64 programas do livro prontos para utilização em seu PC.

CASSETTE OPCIONAL

1. MSX - Melhores Programas. *Software* disponível em invólucro de cassete plastificado, com os 22 "melhores programas" para aplicação imediata no computador MSX e compatíveis. Contém Jogos, Programas Práticos, Pessoais, Educacionais etc..

Diagramação/Composição/Produção:

RioTexto
Tecnologia e Processamento Ltda
Rua Senador Vergueiro, 106/214
Flamengo – Rio de Janeiro – RJ
Tel.: 285-2915 – Cep: 22230

Impresso na  **Prol** editora gráfica Ltda.
03043 Rua Martim Burchard, 246
Brás - São Paulo - SP
Fone: (011) 270-4388 (PABX)
com filmes fornecidos pelo Editor.

dBASE III PLUS

GUIA COMPLETO*

dBASE III Plus – Guia Completo* é uma publicação adaptada para o usuário brasileiro, contendo programas testados e alto índice de depuração, principalmente nas interfaces/interações com programas externos como **Clipper**, **Lotus 1-2-3 v. 2.0**, **SideKick**, **Sideways**, **Turbo Basic** e **WordStar 4**, onde os resultados do leitor/usuário podem ser *realmente* obtidos com o auxílio do livro.

O texto também inclui algumas novidades em termos de apresentação: todos os comandos são comentados na forma de Menus e a partir do *prompt*, via seções “Analisando os Procedimentos”; introduz o conceito de **SuperMacro** (criação do autor!) que define um sistema de programas .PRG (Arquivos de Comando) baseados em um programa de controle central; a estrutura completa de todos os menus – incluindo tradução das descrições dos comandos para o português; Guia de Serviços e Informações Úteis, que fornece os endereços de todos os distribuidores brasileiros dos *softwares* apresentados no livro, bem como dos distribuidores nacionais de livros e revendedores do **disquete opcional** que contém todos os programas do livro prontos para uso com seu dBASE III Plus e muito mais! Excelente pré-requisito para futuros usuários do **dBASE IV**.

* Incluindo estrutura completa dos menus.

dBASE III Plus é marca registrada da Ashton-Tate, representada no Brasil pela Datalógica Sistemas e Comércio de Software Ltda.