

ОСНОВЫ ПРОГРАММИРОВАНИЯ

В СРЕДЕ СУБД FOXPRO.

Саранск
1997

ЦЕНТРОСОЮЗ РОССИЙСКОЙ ФЕДЕРАЦИИ
САРАНСКИЙ КООПЕРАТИВНЫЙ ИНСТИТУТ
МОСКОВСКОГО УНИВЕРСИТЕТА ПОТРЕБИТЕЛЬСКОЙ КООПЕРАЦИИ

ОСНОВЫ ПРОГРАММИРОВАНИЯ В СРЕДЕ СУБД FOXPRO
Методические указания

Саранск
1997

Составители: В.Ф.Байнев, А.А.Сидоров

УДК 681.3

Основы программирования в среде СУБД FoxPro: Метод. указания
/ Сост.: В.Ф.Байнев, А.А.Сидоров; Саран. кооп. ин-т. - Саранск,
1997. - 40 с.

Методические указания содержат начальные сведения по прог-
раммированию в среде одной из самых популярных СУБД для IBM PC -
FoxPro 2.0. Описываются основные команды и способы их применения.
разбираются типовые задачи, решаемые при обработке данных.

Предназначены для студентов различных специальностей, изуча-
ющих СУБД FoxPro.

Печатается по решению учебно-методического совета Саранского
кооперативного института МУПК

ПРЕДИСЛОВИЕ

В 1981 г. фирма IBM выпустила свой первый персональный компьютер IBM PC (Personal Computer), с этого момента началась эпоха компьютеризации, охватившая практически все сферы человеческой деятельности. Благодаря удачной конструкции, обеспечивающей неограниченное расширение возможностей, компьютеры IBM PC и совместимые с ними быстро завоевали популярность у пользователей во всем мире. На рынке программных продуктов появилось большое количество прикладных программ, разработанных специально для IBM PC.

Наряду с улучшением конструкции самого персонального компьютера происходит непрерывное совершенствование его программного обеспечения. Важным шагом в этом направлении стал отказ от повсеместного применения универсальных алгоритмических языков высокого уровня, которые постепенно вытесняются специализированными программными продуктами. Наиболее отчетливо преимущества такого подхода обнаруживаются при использовании прикладных программ, предназначенных для обработки больших объемов данных и получивших название систем управления базами данных (СУБД).

Среди значительного количества СУБД, представленных на рынке программных продуктов, доминирующее положение занимает семейство так называемых dBASE-подобных СУБД, в которое входят такие широко распространенные в России программные продукты, как dBASEIII-plus (РЕБУС), dBASEIV, CLIPPER-5, FoxPro. Последняя, обладая исключительно высокими скоростными характеристиками, заметно выделяется среди всех систем этого класса. Учитывая это обстоятельство, а также то, что в 1991 г. права на производство FoxPro приобрела фирма Microsoft, являющаяся мировым лидером по изготовлению программной продукции, активно работающая на российском рынке, следует ожидать, что данный программный продукт в ближайшее время получит у нас в стране широкое распространение.

1. ОБЩИЕ ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ СУБД FOXPRO

1.1. Структура данных в СУБД FoxPro

Под базой данных в дальнейшем будем подразумевать упорядоченную последовательность наборов данных (записей), описывающих отдельные объекты. Примером базы данных служит городской телефонный справочник, содержащий информацию о фамилии, домашнем адресе и номере телефона каждого абонента телефонной сети. Очевидно, что наиболее наглядно эта информация может быть представлена в виде таблицы, включающей соответствующие графы. Аналогичное представление данных в виде таблицы, принятое в СУБД FoxPro, получило название реляционной модели данных (рис.1.1). При этом каждая строка таблицы, называемая записью, содержит сведения об описываемом объекте. Столбцы таблицы - поля базы данных - несут однотипную информацию. Отдельная клетка любой записи является полем этой записи.

Поле 1	Поле 2	Поле 3	Поле 4	Поле N	
					Запись 1
					Запись 2
					Запись 3
⋮	⋮	⋮	⋮	⋮	⋮
					Запись K

Рис. 1.1. Реляционная модель данных в СУБД FoxPro

В случае телефонного справочника поля соответствующей базы данных могут содержать следующую информацию:

- поле 1 - порядковый номер;
- поле 2 - фамилия и инициалы абонента;
- поле 3 - домашний адрес;
- поле 4 - номер телефона.

При необходимости количество полей в СУБД FoxPro допускается увеличивать. Так, можно ввести поля с информацией о годе рождения абонента, дате установки телефона и т.п. Нетрудно заметить, что поля базы данных различаются не только размером, но и качествен-

ным составом хранящейся в них информации. В соответствии с этим выделяются:

- символьные поля, содержащие в себе набор буквенно-цифровых символов (например, поля фамилий и домашних адресов);
- числовые поля (например, поле года рождения абонента);
- поля дат (например, поле даты установки телефона);
- логические поля.

Полям пользователь должен присвоить имена. Недостатком FoxPro с точки зрения русскоязычного пользователя является то, что имена полей обозначаются латинскими буквами. Примеры имен полей в СУБД FoxPro: Fio, Pol, Data, Naim.

Очевидно, что каждая строка (запись) базы данных описывает отдельный объект. В приведенном примере таким объектом является каждый конкретный телефонный абонент. Отсюда следует, что число записей базы данных должно соответствовать числу абонентов телефонной сети, т.е. числу описываемых объектов.

Вся информация в СУБД FoxPro хранится на магнитных дисках в виде файлов. Как известно, под файлом понимается любой набор данных на диске, имеющий имя. В СУБД FoxPro имя файлу, хранящему базу данных, присваивает пользователь, исходя из принятых в DOS правил. Однако расширение файлов баз данных в СУБД FoxPro стандартное, поэтому имя файла базы данных выглядит следующим образом: <имя файла>.dbf.

Примеры имен файлов баз данных: telefon.dbf, spisok01.dbf, sklad.dbf, svetlana.dbf.

Файлы баз данных (dbf-файлы) являются основными носителями данных на диске. Они имеют следующие характеристики:

- число записей в файле - до 1 млрд;
- размер записи (байт) - до 4000;
- число полей в записи - до 255;
- число одновременно открытых баз - до 25.

Максимальные размеры полей СУБД FoxPro также ограничены и соответствуют значениям (байт):

- символьные поля, тип C (Character) - до 254;
- числовые поля, тип N или F (Numeric или Float) - до 20;
- поля дат, тип D (Date) - 8;
- логические поля, тип L (Logical) - 1.

Примечание. Байтом называется единица информации, достаточ-

ная для отображения одного символа (буквы, цифры, знака препинания).

Как следует из сказанного, поля в СУБД FoxPro имеют ограничения максимального размера. Однако на практике часто бывает необходимо разместить в записи информацию, объем которой превышает максимально допустимый для указанных полей. В этом случае база данных может быть расширена за счет так называемых полей примечаний (Мето-полей). Эти поля (тип М) имеют произвольную длину в каждой записи и практически ничем не ограничены. Они создаются одновременно с остальными полями и совершенно равноправны с ними, однако хранятся не в dbf-файле, а в специальном связанном с ним файле примечаний. Файл примечаний имеет одинаковое с файлом базы данных имя и расширение fpt. Например, если основные поля базы данных, содержащей сведения об абонентах телефонной сети, хранятся в файле telefon.dbf, то Мето-поля, при их наличии, - в файле с именем telefon.fpt.

ЗАДАНИЕ 1.1

Составьте на листе бумаги базу данных на членов вашей семьи, содержащую следующие поля: порядковый номер, фамилия и инициалы, дата рождения, пол, возраст (лет). Определите типы этих полей, назначьте им имена и размеры.

1.2. Синтаксис команд СУБД FoxPro

Запуск СУБД FoxPro средствами Norton Commander осуществляется путем выделения курсором файла - загрузочного модуля foxpro1.exe, находящегося в директории FOXPRO, и нажатия на клавишу Enter. После появления начальной заставки на экране открывается так называемое командное окно со словом Command, приглашающее пользователя ввести команды управления СУБД. Команды набираются пользователем в окне при помощи клавиатуры. Общая длина команды не должна превышать 2048 символов (включая пробелы и знаки препинания). Для переноса в тексте программы части команды на следующую строку в конце текущей строки ставится знак ";".

Перечислим в порядке приоритетов знаки операций, которые могут входить в команды.

Математические:

- 1) ** - возведение в степень;
- 2) * - умножение, / - деление, % - остаток от деления;
- 3) + - сложение, - - вычитание.

Логические:

- 1) NOT - логическое отрицание НЕ;
- 2) AND - логическое И;
- 3) OR - логическое ИЛИ.

Знаки логических операций с обеих сторон выделяются точками.
Например: .AND., .OR., .NOT.

Отношения:

- < - меньше; > - больше; = - равно; # - не равно;
<= - не больше; >= - не меньше.

При описании команд будем использовать следующие обозначения:

[...] - в квадратных скобках указывается та часть команды, которая может в ней отсутствовать. Скобки в команду не входят;

<...> - в угловых скобках помещается всякое разрешенное выражение, которое программист должен поместить в команду, скобки в нее не входят;

.../... - знак говорит о том, что в команде необходимо наличие только одного из элементов, разделенных этим знаком;

ВыrN - выражение числового типа (Numeric). Его результатом является число;

ВыrC - выражение символьного типа (Character). Ему соответствует набор символов или отдельный символ;

ВыrD - выражение типа "дата" (Date). Его результатом является число, к которому применима арифметика дат.

Общее число команд СУБД FoxPro достигает пятисот, однако все они имеют общую структуру. Команды FoxPro, ориентированные на обработку файлов баз данных, могут быть представлены так:

```
НАЗВАНИЕ [<границы>] [<список выражений>];  
[FOR <условие>] [WHILE <условие>]
```

Здесь НАЗВАНИЕ - имя команды;

[<границы>] - границы действия команды, которые могут иметь одно из следующих значений:

- ALL - все записи базы данных;
REST - все записи, начиная с текущей;

NEXT <N> - следующие N записей, начиная с текущей;
RECORD <N> - запись номер N;
[FOR <условие>] - выполнение команды только для записей, отвечающих <условию>;
[WHILE <условие>] - выполнение команды только до тех пор, пока не перестанет выполняться <условие>.

Примечание. Текущей записью считается та, в которой в данный момент находится курсор.

Кроме того, в СУБД FoxPro имеется большая группа команд, которые не влекут за собой каких-либо немедленных действий, но определяют условия работы других команд. Такие команды называются командами-установками и имеют следующую структуру:

SET <параметр команды> TO <значение параметра>
SET <параметр команды> OFF/ON

В этих случаях <параметр команды> может быть задан некоторым <значением> или включен/выключен (ON/OFF).

Пример. Команда SET BELL OFF позволяет отключить звуковой сигнал компьютера. Аналогичная команда с ключом ON вновь включает его.

В FoxPro в составе команд разрешается использовать как прописные, так и строчные буквы. Это, конечно, не относится к данным, где строчные и прописные буквы неэквивалентны, поскольку имеют разные коды представления в компьютере.

В состав команд могут быть включены константы, которые так же, как и типы полей, бывают символьными, числовыми, логическими и константами типа "дата".

Символьные константы выделяются апострофами, кавычками или квадратными скобками. Примеры символьных констант:

'Петров А.А.', "9-94-49", [телевизор "Sony"].

Константы типа "дата" берутся в фигурные скобки. Пустая дата может быть задана одними лишь фигурными скобками. Примеры констант типа "дата":

{15.11.95}, {04.04.66}, {07.10.94}.

Логические константы могут иметь только два значения: T - TRUE (истина) и F - FALSE (ложь). В тексте программ они обрамляются точками: .T., .t., .F., .f.

Смысл применения констант заключается в следующем. Предположим, что в базе данных телефонных абонентов вам необходимо найти

номер телефона, принадлежащего Абрашкиной Л.И. Для этого вы в качестве символьной константы используете строку 'Абрашкина Л.И.', с которой сравниваете поля фамилий всех записей базы данных до тех пор, пока анализируемое поле полностью не совпадет с заданной константой. Аналогично можно осуществить поиск и по телефонному номеру. В этом случае символьной константой будет номер телефона.

Ниже приводится описание команд, позволяющих пользователю создать базу данных, заполнить ее, провести редактирование уже имеющихся баз данных.

Примечание. Освоение команд FoxPro желательно производить непосредственно у компьютера, опробуя их в интерактивном режиме.

2. СОЗДАНИЕ ФАЙЛА БАЗЫ ДАННЫХ

Создание файла базы данных включает в себя два основных этапа: образование структуры файла и его заполнение данными, что соответствует логике работы с таблицами. Действительно, сначала мы определяем количество граф, задаем заголовок и размер каждой графы, расчерчиваем таблицу, т.е. определяем ее структуру, после чего приступаем к ее заполнению данными.

2.1. Создание структуры файла базы данных

Для создания структуры файла базы данных в FoxPro существует специальная команда:

```
CREATE <имя файла>
```

Действие команды проиллюстрируем конкретным примером. Предположим, что для учета товаров на складе должна быть создана база данных с именем sklad01.dbf, содержащая следующие сведения о каждом товаре.

- | | |
|----------------------------------|---------|
| 1. Наименование товара | (Naim). |
| 2. Дата поступления на склад | (Dtp). |
| 3. Количество единиц товара | (Kol). |
| 4. Цена за единицу | (St1). |
| 5. Общая стоимость | (Stm). |
| 6. Краткая характеристика товара | (Prim). |

В скобках указаны названия полей (в каждом конкретном случае они выбираются пользователем, поэтому могут выглядеть и по-другому).

му). В поле с именем Prim будем заносить информацию о характеристике товара. Определим для полей базы данных их типы и размеры.

1. Naim - символьный тип (Character) длиной до 25 символов.
2. Dtp - тип "дата" (Date) со стандартной длиной 8 символов.
3. Kol - числовой тип (Numeric) длиной 4 разряда целых.
4. St1 - числовой тип (Numeric) длиной 8 разрядов целых.
5. Stm - числовой тип (Numeric) длиной 12 разрядов целых.
6. Prim - тип Memo-поле (ввиду того что для разных видов товара объем информации в этом поле может иметь различное значение, выбираем для нее поле примечаний).

После запуска СУБД FoxPro пользователь набирает:

```
CREATE sklad01
```

Примечание. Расширение файла базы данных указывать не обязательно, поскольку СУБД FoxPro подразумевает, что названный файл по умолчанию имеет расширение dbf.

В ответ на эту команду СУБД представит экран-форму для ввода данных о структуре создаваемого файла, а именно: для каждого вводимого поля - его имя (Name), тип (Type), длину (Width) и, для числового поля, точность (Dec), интерпретируемую как число десятичных знаков после запятой. Для файла sklad01.dbf заполним предлагаемую форму в соответствии с рис. 2.1. следующим образом. Сначала при помощи клавиатуры вводится имя поля. Тип поля выбирается в момент, когда курсор находится в колонке Type, нажатием первой буквы указания типа (C,N,F,D,L,M) или клавиш Space/Enter (Пробел/Ввод). Во втором случае в ответ появляется меню выбора типа поля (рис. 2.2), в котором курсором и нажатием клавиши Enter осуществляется выбор типа текущего поля. Перемещение курсора свободно осуществляется как внутри колонок по вертикали, так и между ними по горизонтали с помощью предназначенных для этого клавиш. В нижней части формы определения структуры ведется статистика. Как видно из рис. 2.1, база данных содержит 6 полей (Fields) общей длиной (Length) 68 байт. Еще доступно для заполнения (Available) в текущей записи 3932 байт (из 4000 возможных).

Structure: D:\FOXPRO\SKLADO1.dbf

Name	Type	Width	Dec	Field
NAIM	Character	25		
DTP	Date	8		
KOL	Numeric	4	0	<Insert>
ST1	Numeric	8	0	<Delete>
STM	Numeric	12	0	<OK>
PRIM	Memo	10		<Cancel>

Fields: 6 Length: 68 Available: 3932

Character
Numeric
Float
Date
Logical
Memo

Рис. 2.2.
Меню выбора
типа поля

Рис. 2.1. Форма ввода данных о структуре dbf-файла

ВНИМАНИЕ! По окончании формирования структуры файла она должна быть сохранена на диске. Это осуществляется одновременным нажатием клавиш Ctrl+W или Ctrl+End. Нажатие клавиши Escape вызовет отказ от сохранения структуры. Тот же результат может быть достигнут нажатием клавиши Enter после перемещения курсора в позицию <OK> или <Cancel> соответственно.

После того как структура базы данных будет создана, СУБД запросит ввод данных:

Input data records now?
<Yes>
<No>

Ответим <No>.

Если в дальнейшем обнаружится, что структура базы данных нас не удовлетворяет, ее можно будет изменить командой модификации структуры

MODIFY STRUCTURE

В результате пользователь попадет в меню, идентичное меню команды CREATE, в котором можно удалять, дополнять, переименовывать поля базы данных, а также изменять их параметры. При этом модифицируемый файл должен быть предварительно открыт (см. далее по тексту команду USE).

ЗАДАНИЕ 2.1

Загрузите СУБД FoxPro по правилам, изложенным в п.1.2. Используя команду CREATE, создайте структуру файла базы данных sklad01.dbf. Задайте имена, типы и размеры полей в соответствии с информацией, приведенной в п. 2.1. После заполнения формы рис. 2.1 осуществите запись на диск созданной структуры файла нажатием клавиш Ctrl+W или Ctrl+End и выйдите из среды СУБД FoxPro в DOS, набрав в командном окне команду QUIT. Выход в DOS может быть осуществлен иначе: из главного меню СУБД (верхняя строка). Для этого нажатием клавиши F10 активизируйте главное меню. Клавишами перемещения курсора выделите пункт меню File. В открывшемся окне после нажатия клавиши Enter выделите пункт Quit (Выход) и выйдите из среды СУБД FoxPro нажатием на Enter. Возвратившись в DOS, убедитесь, что в результате ваших действий в директории FOXPRO появился новый файл базы данных с именем sklad01.dbf.

2.2. Заполнение базы данных

Файл после создания структуры остается открытым, т.е. доступным для команд ввода, просмотра и изменения. Однако, если СУБД FoxPro только что загружена в память, должно быть выполнено открытие нужного файла базы данных командой открытия

USE [<имя файла>]

Команда USE без имени файла закрывает базу данных. Закрытие всех файлов баз данных и связанных с ними файлов других типов (например, fpt-файлов) осуществляется командой

CLOSE DATABASE

Закрытие вообще всех файлов выполняется командой

CLOSE ALL

После того как откроется файл базы данных, его можно дополнить новыми записями (в том числе, если он еще совсем пуст). Дополнение файла новыми записями осуществляется командой

APPEND

Она предьявляет окно ввода данных со всеми пустыми полями заполняемой записи. Для созданного при выполнении задания 2.1 файла базы данных это окно будет иметь вид такой, как представлен на рис. 2.3.

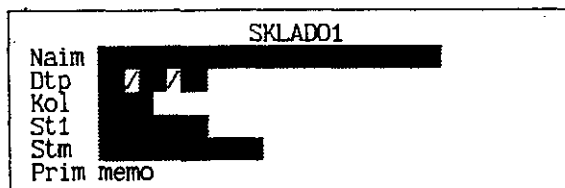


Рис. 2.3. Окно ввода данных

Все поля базы данных выделяются контрастным цветом в соответствии с выбранной при создании структуры базы данных длиной (см. рис. 2.1). Данные могут быть введены в эти поля непосредственно с клавиатуры. После ввода текущей записи автоматически появляется доступ к следующей записи, и так до тех пор, пока не будут заполнены поля последней записи. Для того чтобы войти в Мемо-поле, необходимо поместить в него курсор и нажать Ctrl+Home. После этого в него вводится необходимая буквенно-цифровая информация. Выход из Мемо-поля с сохранением сделанных в нем изменений на диске осуществляется клавишами Ctrl+W или Ctrl+End, без сохранения - Escape.

Как известно, в разных странах мира принят различный формат представления даты. По умолчанию в FoxPro представляется американский формат даты - две цифры месяца, дня и года, разделенные косой чертой, т.е. ММ/ДД/ГГ. Именно в таком виде предполагается ввод даты в окне на рис. 2.3. Очевидно, что для использования в России и в некоторых других странах этот формат неудобен, поэтому имеется возможность изменить его одной из указанных ниже команд (справа приведены примеры написания даты 5 декабря 1995 г. в соответствующих форматах):

SET DATE AMERICAN	ММ/ДД/ГГ	12/05/95
SET DATE ANSI	ГГ.ММ.ДД	95.12.05
SET DATE BRITISH/FRENCH	ДД/ММ/ГГ	05/12/95
SET DATE GERMAN	ДД.ММ.ГГ	05.12.95
SET DATE ITALIAN	ДД-ММ-ГГ	05-12-95

Поскольку для нас привычным является германский формат представления даты, то в командном окне сразу же после команды открытия файла целесообразно использование команды SET DATE GERMAN.

ЗАДАНИЕ 2.2

Загрузите СУБД FoxPro. В командном окне осуществите выполнение последовательности команд:

```
USE sklad01
SET DATE GERMAN
APPEND
```

Примечание. Выполнение этого задания возможно только после того, как будет выполнено задание 2.1, т.е. при наличии на диске файла базы данных sklad01.dbf.

Заполните поля открытой базы данных в соответствии с рис. 2.4.

Naim	Dtp	Kol	St1	Stm	Prim
Телевизор "Sony"	01.12.94	12	2560000	30720000	Мето
Велосипед "Кросс"	16.02.95	25	250000	6250000	мето
В/магнитофон "Sanyo"	22.06.95	4	1200000	4800000	Мето
Дубленка женская	22.06.95	10	2500000	25000000	мето
П/ботинки мужские	25.10.95	125	230000	28750000	Мето
В/плейер "Akai"	25.10.95	14	990000	13860000	мето
Люстра "Элегия"	30.11.95	72	455500	32796000	мето
Р/приемник "Нейва"	02.12.95	170	187000	31790000	мето
Бра "Интим"	02.12.95	22	220000	4840000	мето
Холодильник "Наст"	25.01.96	8	3350000	26800000	мето

Рис. 2.4. Пример заполнения базы данных

На рис. 2.5 приведен пример заполнения окна ввода данных для первой записи этой базы данных. Обратите внимание на то, что форма представления даты в данном случае иная, чем на рис. 2.3.

SKLADO1	
Naim	Телевизор "Sony" ██████████
Dtp	01.12.94
Kol	12
St1	2560000
Stm	30720000
Prim	мето

Рис. 2.5. Пример заполнения первой записи базы данных

Для первой, третьей и пятой записей введите следующую текстовую информацию в Мето-поля:

- а) для первой: "Размер экрана по диагонали 67 см";
- б) для третьей: "Мультисистемный PAL/SECAM/NTSC";
- в) для пятой: "Размеры 26,5 и 27. Цвет черный".

Для остальных записей Мемо-поля оставьте пустыми. Обратите внимание на то, что информация в Мемо-полях не отображается, а само Мемо-поле обозначается словом "Мемо". Если оно начинается со строчной буквы, то поле пустое, если с прописной ("Мемо") - в нем имеются данные. После ввода всех десяти записей сохраните введенные данные нажатием Ctrl+W или Ctrl+End. Осуществите выход из СУБД FoxPro и убедитесь в наличии на диске в директории FOXPRO файла базы данных sklad01.dbf и связанного с ним файла примечаний sklad01.fpt.

Примечание. Ввод русских букв возможен с клавиатуры при использовании драйвера-русификатора, который, как правило, находится в памяти компьютера как резидентная программа. В зависимости от типа русификатора осуществляется переключение на русский регистр клавишами RightShift или Ctrl, обратно на латинский регистр - клавишами LeftShift или Ctrl (повторно). Для некоторых типов русификатора переключения осуществляются нажатием на обе клавиши LeftShift+RightShift.

3. УПРАВЛЕНИЕ БАЗАМИ ДАННЫХ

Под управлением базами данных можно подразумевать выполнение следующих операций: редактирование баз данных, перемещения в базе данных, просмотр, удаление и изменение данных, локализация и поиск данных в базе, одновременная работа с несколькими базами данных, ввод-вывод данных, копирование баз данных.

3.1. Редактирование баз данных

При выдаче команд APPEND, INSERT, EDIT, CHANGE, BROWSE и при наличии открытой базы данных FoxPro разворачивает для пользователя окно редактирования. Вся информация в этом окне доступна для изменения, в том числе возможны как удаление, так и дополнение новых записей.

Стандартное окно редактирования имеет две формы. Для первых четырех команд оно будет выглядеть совершенно одинаковым - все

поля базы данных располагаются вертикально (аналогично команде APPEND). На экране видно столько записей и полей, сколько удастся их здесь разместить. Такая форма окна редактирования называется CHANGE-окном. Возможно и более привычное для пользователя представление данных, когда все поля каждой записи располагаются горизонтально (в виде обычной таблицы). Такая форма представления данных получила название BROWSE-окна. Если какие-то поля записи не уместятся в строке, то с помощью клавиш управления курсором осуществляется свободное перемещение (скроллинг) изображения вправо и влево. Клавиши управления курсором действуют практически одинаково в обеих формах окон. Кроме привычных для пользователя клавиш управления курсором при редактировании часто используются следующие клавиши:

Home/End	- переход к началу/концу поля;
Tab/Shift+Tab	- переход к следующему/предыдущему полю;
PgUp/PgDn	- переход назад/вперед на один экран;
Enter	- переход к следующему полю;
Ctrl+Y	- удаление поля;
Ctrl+Home, Ctrl+PgUp,	
Ctrl+PgDn	- вход в Метод-поле;
Ctrl+W или Ctrl+End	- выход с сохранением измененных данных;
Escape	- выход без сохранения изменений в текущем окне.

Дополнение базы данных новой записью осуществляется нажатием клавиш Ctrl+N (кроме команды APPEND, в которой это происходит автоматически). Нажатием Ctrl+T производится пометка записи к удалению. Признаком пометки служит появление точки в самой левой колонке записи.

BROWSE-окно

Команда BROWSE - один из наиболее мощных и удобных инструментов доступа пользователя к данным СУБД FoxPro. В результате выполнения этой команды записи базы данных располагаются на экране горизонтально. Их можно редактировать, дополнять и помечать к удалению. Кроме того, команда BROWSE позволяет настроить удобную форму предъявления данных на экране как из одной, так и одновре-

менно из нескольких баз данных. Возможность создания вычисляемых полей, являющихся функциями других полей базы данных, придает СУБД FoxPro свойства, присущие электронным таблицам.

Упрощенный формат команды:

```
BROWSE [FIELDS <поля>] [FOR <условие>];
      [FREEZE <поле>] [NOAPPEND];
      [NOEDIT/NOMODIFY] [NODELETE];
      [TITLE <ВырС>] [LOCK <ВырN1>];
      [PARTITION <ВырN2>] [NOLINK]
```

Рассмотрим действие опций команды.

FIELDS <поля> - перечень выводимых полей. Если эта опция отсутствует, то командой BROWSE предъявляются все поля базы данных. Имя каждого поля может сопровождаться следующими ключами:

:R - разрешен только просмотр поля;

:<ВырN3> - видимый размер поля в BROWSE-окне;

:B=<Выр1>,<Выр2> - указание границ чисел и дат;

:N=<ВырC1> - указание собственного заголовка поля, который выносится в "шапку" таблицы;

:P=<ВырC2> - задание формата отображения данных.

В список полей опции FIELDS могут включаться вычисляемые поля, которые являются функциями других полей базы данных. Например, при отображении полей базы данных (рис. 2.4) возможен вывод на экран дополнительного (седьмого) поля, несущего информацию о стоимости товара в долларах США, вычисляемую как частное стоимости товара в рублях и курса доллара.

FOR <условие> - устанавливает фильтр для записей базы. В BROWSE-окне размещаются только записи, удовлетворяющие заданному <условию>.

Пример.

```
USE sklad01
BROWSE FOR St1<=1000000
```

В данном примере команда BROWSE предъявляет только те записи базы данных sklad01.dbf, в которых значение поля St1 (цена за единицу) не превышает 1000000 р.

FREEZE <поле> - указывает единственное поле, значение которого разрешается редактировать.

NOAPPEND - дополнение базы с помощью клавиш невозможно.

NOEDIT/NOMODIFY - редактирование невозможно. Разрешены пометка к удалению и дополнение базы.

NODELETE - пометка к удалению записей базы с помощью клавиш Ctrl+T невозможна.

TITLE <ВырС> - задает заголовок <ВырС> BROWSE-окна.

LOCK <ВырN1> - BROWSE-окно делится на две части, где первые <ВырN1> полей всей базы или из списка FIELDS дублируются в левой части окна. Переход курсора из одной половины окна в другую осуществляется нажатием клавиш Ctrl+N.

PARTITION <ВырN2> - то же, причем граница будет проходить по колонке <ВырN2>.

NOLINK - несинхронное перемещение видимых записей в разделенных окнах. При движении курсора в одной половине окна записи в другой половине остаются неподвижными.

ЗАДАНИЕ 3.1

Загрузите СУБД FoxPro. Выполните в командном окне команды:

```
USE sklad01
SET DATE GERMAN
BROWSE
```

Рассмотрите форму предъявления данных в BROWSE-окне (она соответствует рис.2.4). Нажмите на клавиши Ctrl+F10. Это позволит "распахнуть" окно на весь экран монитора. Повторное нажатие указанной комбинации клавиш возвратит экран в исходное состояние. Нажатием клавиши Escape вернитесь в командное окно. Для оценки возможностей команды BROWSE осуществите в командном окне выполнение команды со следующими опциями:

```
BROWSE TITLE 'ПЕРЕЧЕНЬ ТОВАРОВ НА СКЛАДЕ' ;
FIELDS Naim :H='Наименование товара', ;
      Dtp :H='Дата', ;
      Kol :H='Кол.', ;
      St1 :H='Цена,р.', ;
      Std=St1/4500 :H='Цена,$' :P='###.##' ;
FOR St1<=1000000
```

Внимательно рассмотрите таблицу, предъявляемую на экран командой. Опция TITLE задает заголовок таблице. Опция FIELDS позволяет вывести на экран не все, а только перечисленные поля Naim, Dtp, Kol и St1. Кроме того, выводится вычисляемое поле с именем Std, несущее информацию о цене товара в долларовом выражении (при

расчете 4500 р. : 1 дол.). Ключи :H=' задают названия соответствующим колонкам таблицы, ключ :P~'####.##' - шаблон вывода числовой информации - шесть десятичных знаков, из которых два знака - после запятой. Условие FOR St1<=1000000 обеспечивает представление только тех записей, у которых значение поля St1 (цена за единицу) не превышает 1000000.

Выйдите из BROWSE-окна нажатием клавиши Escаре.

Примечание. Выполнение этого задания (как и последующих) возможно только в случае корректного выполнения предыдущего.

CHANGE-окно

Команда CHANGE предъявляет на редактирование перечисленные поля записей базы данных в указанных границах и для заданных условий.

```
CHANGE [<границы>] [FOR <условие>];  
        [WHILE <условие>] [FIELDS <поля>] [опции]
```

<Границы>, FOR- и WHILE-условия ограничивают работу команды только заданным диапазоном записей (см. п.1.2). Отсутствие границ и условий означает, что доступны все записи базы данных. FIELDS <поля> - перечень обрабатываемых полей. Отсутствие ключевого слова FIELDS указывает на обработку всех полей записи. Опции - набор режимов по составу и действию почти полностью аналогичных опциям команды BROWSE.

Примечание. Команда EDIT полностью идентична команде CHANGE.

ЗАДАНИЕ 3.2

Загрузите СУБД FoxPro. Осуществите в командном окне выполнение следующих команд:

```
USE sklad01  
SET DATE GERMAN  
CHANGE FIELDS Naim, Dtp, St1 ;  
FOR Dtp>={01.01.95}.AND.Dtp<{01.01.96}
```

Рассмотрите выводимую на экран монитора таблицу. Как видно, команда предъявляет для редактирования только три поля: Naim, Dtp и St1. Кроме того, на экране размещаются не все записи базы данных, а только те, для которых выполняется условие

RECALL [<границы>] [WHILE <условие>] [FOR <условие>] - снятие пометок к удалению. RECALL без параметров действует только на текущую запись.

ЗАДАНИЕ 3.3

Используя средства Norton Commander, скопируйте в директорию D:\FOXPRO файл sklad01.dbf под именем sklad02.dbf и файл sklad01.fpt под именем sklad02.fpt. Это позволит вам иметь на диске два файла баз данных, один из которых мы в дальнейшем будем изменять.

Загрузите СУБД FoxPro. В новой базе данных sklad02.dbf удалите все "старые" записи, сделанные до 1 января 1995 г., организовав просмотр удаляемых записей. Осуществите в командном окне выполнение следующих команд:

```
USE sklad02
SET DATE GERMAN
SET MEMOWIDTH TO 20
CLEAR           && Команда очистки экрана
DISPLAY Naim, Dtp, Kol, Prim FOR Dtp<{01.01.95}
```

ВНИМАНИЕ! Нажатие Ctrl+F9 позволяет скрыть/раскрыть командное окно

```
DELETE FOR Dtp<{01.01.95}
PACK
BROWSE
```

Команда DISPLAY выводит на экран указанные в ней поля базы данных sklad02.dbf, для которых значение поля Dtp<{01.01.95}, т.е. на экран предъявляются записи, сделанные в 1994 г. и ранее. Обратите внимание на то, что Метод-поле в данном случае представляется не в виде колонки с надписью "метод", а явно. Фактическая ширина этого поля в 20 символов устанавливается командой SET MEMOWIDTH TO 20. Текст, набранный после знаков &&, никак не воспринимается СУБД и может использоваться в качестве комментария к тексту программы.

В FoxPro имеется возможность не только вручную редактировать данные, но и изменять их путем присваивания и вычислений при помощи команды

```
REPLACE [<границы>] [WHILE <условие>] [FOR <условие>] ;  
<поле1> WITH <выражение> [,<поле2> WITH <выражение>...] ;  
[ADDITIVE] [NOOPTIMIZE]
```

Эта команда осуществляет множественное изменение полей базы данных в соответствии с заданными выражениями, в установленных границах и при заданных условиях. Если отсутствует параметр <границы> или <условия>, то будет изменена только текущая запись. Параметр ADDITIVE действует для Memo-полей и означает, что заданное <выражение> будет дописываться в конце поля. Если этот параметр опущен, то старое значение Memo-поля будет замещено <выражением>.

ЗАДАНИЕ 3.4

В связи с изменением цен на товары осуществите коррекцию полей St1 и Stm базы данных sklad02.dbf в соответствии со следующими условиями.

1. Если цена товара превышает 2 млн р., то его цена снижается на 10 %.

2. Если цена товара превышает 1 млн р., но составляет величину не более 2 млн р., то она снижается на 5 %.

3. Для остальных товаров цена остается прежней.

Для осуществления этого задания необходимо выполнить в командном окне последовательность команд:

```
USE sklad02  
SET DATE GERMAN  
REPLACE FOR St1>2000000 ;  
St1 WITH St1*0.9, Stm WITH Stm*0.9  
REPLACE FOR St1>1000000.AND.St1<=2000000 ;  
St1 WITH St1*0.95, Stm WITH Stm*0.95  
BROWSE
```

Рассмотрите предъявленную на экран таблицу (рис. 3.1) и сравните ее с рис.2.4 (первая запись удалена в результате выполнения задания 3.3). Командами REPLACE осуществлены необходимые изменения значений полей St1 и Stm.

Наим	Dtp	Kol	St1	Stm	Prim
Велосипед "Кросс"	16.02.95	25	250000	6250000	memo
В/магнитофон "Sanyo"	22.06.95	4	1140000	4560000	Memo
Дубленка женская	22.06.95	10	2250000	22500000	memo
П/ботинки мужские	25.10.95	125	230000	28750000	Memo
В/плеер "Akai"	25.10.95	14	990000	13860000	memo
Люстра "Элегия"	30.11.95	72	455500	32796000	memo
Р/приемник "Нейва"	02.12.95	170	187000	31790000	memo
Бра "Интим"	02.12.95	22	220000	4840000	memo
Холодильник "Наст"	25.01.96	8	3015000	24120000	memo

Рис. 3.1.. Вид экрана после выполнения задания 3.4

ЗАДАНИЕ 3.5

Выведите на принтер из базы данных sklad01.dbf сведения о товарах, поступивших в 1995 г., стоимостью не более 750 тыс. р. Для этого осуществите выполнение в командном окне последовательности команд:

```
USE sklad01
SET DATE GERMAN
CLEAR
DISPLAY FOR Dtp<{01.01.96}.AND.Dtp>={01.01.95};
.AND.St1<=750000
DISPLAY FOR Dtp<{01.01.96}.AND.Dtp>={01.01.95};
.AND.St1<=750000 TO PRINT
```

Если к моменту выполнения последней команды задания принтер окажется не готовым к работе, то на экране дисплея появится сообщение

Printer not ready. Retry? (Принтер не готов. Повторить?) <Yes> <No>
--

В этом случае необходимо подготовить принтер к работе и ответить <Yes>.

Рассмотрите полученную распечатку. Обратите внимание на то, что в команде DISPLAY условие для опции FOR составлено из трех простых условий, объединенных знаками логического "И". В связи с отсутствием в команде опции OFF записи были пронумерованы. Первая

команда DISPLAY осуществила вывод данных не на принтер, а на экран дисплея, что позволило предварительно просмотреть выводимые на печать результаты.

3.4. Локализация и поиск данных в базе

Так как одной из наиболее распространенных операций при использовании СУБД является поиск данных в базе, то группа команд для отбора (локализации) и поиска данных в FoxPro имеет исключительное значение. Хотя в некоторых ранее рассмотренных командах (BROWSE, CHANGE, DISPLAY) предусматривается отбор записей с помощью опции FOR, в других командах это может оказаться невозможным или неудобным. Команда

```
SET FILTER TO [<условие>]
```

позволит установить FOR-условие для всех без исключения команд обработки данных в активной базе данных.

В FoxPro имеется несколько команд, позволяющих осуществлять поиск данных, причем он может быть как последовательным, так и ускоренным. При последовательном поиске выполняется сплошной перебор записей файла базы данных до тех пор, пока не будет найдена искомая. В случае ускоренного поиска используется так называемое индексирование базы данных (об индексировании будет рассказано отдельно).

Для осуществления последовательного поиска в базе данных самой первой записи, удовлетворяющей заданному FOR-условию, среди записей, находящихся в заданных границах, и до тех пор, пока соблюдается WHILE-условие, существует команда

```
LOCATE FOR <условие> [<границы>] [WHILE <условие>]
```

При успешном поиске указатель записей устанавливается на искомую запись и процесс поиска прекращается. Продолжить начатый поиск можно командой

```
CONTINUE
```

По окончании поиска на экране дисплея появляется надпись "End of Locate scope" ("Конец границы поиска").

ЗАДАНИЕ 3.6

В базе данных sklad01.dbf найдите запись, хранящую информа-

цию о люстре "Элегия", и выведите ее на принтер. Для этого необходимо выполнить в командном окне СУБД FoxPro последовательность команд:

```
USE sklad01
LOCATE FOR Naim='Люстра "Элегия"'
CLEAR
DISPLAY TO PRINT
CONTINUE
```

ВНИМАНИЕ! При использовании в составе команды текстовой константы 'Люстра "Элегия"' обратите особое внимание на правильность написания этой константы. При поиске компьютер будет осуществлять проверку на ее полное совпадение с информацией, содержащейся в поле Naim, включая наличие пробелов между словами, знаков препинания, а также учитывая различия между строчными и прописными буквами. Например, если в рассматриваемой текстовой константе после буквы "а" будет поставлено более одного пробела или первая буква "Л" будет введена как строчная, то искомая запись не будет найдена.

Так как команда LOCATE устанавливает указатель записей на искомую запись, делая ее текущей, а команда DISPLAY без параметров действует только для текущей записи, то именно искомая запись будет отражена на экране дисплея и выведена на печать. Для того чтобы убедиться в том, что искомая запись единственная, поиск можно продолжить командой CONTINUE.

3.5. Индексирование баз данных

Важнейшим элементом любой СУБД является наличие средств ускоренного поиска данных, поскольку в случае больших баз данных последовательный поиск осуществляется крайне медленно. Одним из наиболее эффективных средств ускоренного доступа к данным служит индексирование баз данных. Смысл индексирования заключается в том, что для файла базы данных создается специальный так называемый индексный файл, имеющий расширение idx или cdx. Если файл проиндексирован по какому-либо полю, то все команды, связанные с движением в файле базы данных, перемещают указатель записей не в порядке физического расположения записей в dbf-файле путем их последовательного перебора, а в соответствии с индексным файлом.

Так как индексный файл "упорядочивает" записи dbf-файла по какому-либо полю, то поиск данных по хранящейся в этом поле информации может быть осуществлен гораздо быстрее. Так, например, если проиндексировать базу данных sklad01.dbf по полю St1 (стоимость единицы товара), то поиск товара с заданной ценой будет осуществляться намного быстрее по сравнению со случаем последовательного поиска. Необходимо отметить, что один и тот же dbf-файл может быть проиндексирован по нескольким полям и иметь любое число индексных файлов. Такие файлы содержат информацию о расположении записей файла базы данных в алфавитном, хронологическом или числовом порядке для того поля (полей), по которому выполнено индексирование.

В FoxPro можно создать два типа индексных файлов.

1. Обычный индексный файл. Он имеет расширение имени idx, содержит один индексный ключ и называется одноиндексным файлом.

2. Мультииндексный файл. Такой файл имеет расширение cdx, хранит сразу несколько индексных выражений и является по существу соединением нескольких обычных индексных файлов.

Индексирование открытой базы данных можно осуществить командой

```
INDEX ON <Выр> TO <idx-файл>/TAG <имя тега> [OF <cdx-файл>];  
[FOR <условие>] [COMPACT] [DESCENDING] [ADDITIVE]
```

Опции команды:

<Выр> - индексный ключ-выражение. Его длина может достигать 100 символов для idx-файлов и 254 для cdx-файлов. Обычно ключ означает имя поля, по которому нужно упорядочить файл. Ключ может быть и составным из нескольких полей (для мультииндексного файла);

TO <idx-файл> - задает имя одноиндексному файлу (расширение можно не указывать, так как подразумевается, что оно idx);

TAG <имя тега> [OF <cdx-файл>] - задает имя тега в мультииндексном файле. Тегом называется отдельный индекс мультииндексного файла. Если в команде присутствует фраза OF <cdx-файл>, то мультииндексный файл получит указанное имя. В противном случае будет создан мультииндексный файл с именем, совпадающим с именем файла базы данных, но имеющий расширение cdx;

FOR <условие> - устанавливает режим отбора в индексный файл только тех записей базы данных, которые удовлетворяют условию;

COMPACT - с этой опцией будет создан компактный idx-файл. Целесообразно использовать только компактные индексные файлы, так как они занимают меньше места на диске;

DESCENDING - индексирование будет выполнено по убыванию. Этот режим можно указывать только для idx-файлов. В случае отсутствия этой опции индексирование выполняется по возрастанию. Для idx-файлов индексирование всегда выполняется по возрастанию;

ADDITIVE - вновь создаваемые индексные файлы не закроют уже открытые к этому моменту. По умолчанию вновь создаваемые индексные файлы закрывают все ранее открытые для текущей базы данных.

ЗАДАНИЕ 3.7

Упорядочите базу данных sklad01.dbf в порядке возрастания цены, т. е. проиндексируйте ее по полю St1. Для этого необходимо создать индексный файл sklad01.idx набором команд:

```
USE sklad01
SET DATE GERMAN
INDEX ON St1 TO sklad01 COMPACT
LIST Naim, St1, Kol, Dtp TO PRINT
```

Результатом выполнения этих команд будет распечатка, представленная на рис. 3.2.

Record #	Naim	St1	Kol	Dtp
8	Р/приемник "Нейва"	187000	170	02.12.95
9	Бра "Интим"	220000	22	02.12.95
5	П/ботинки мужские	230000	125	25.10.95
2	Велосипед "Кросс"	250000	25	16.02.95
7	Люстра "Элегия"	455500	72	30.11.95
6	В/плеер "Akai"	990000	14	25.10.95
3	В/магнитофон "Sanyo"	1200000	4	22.06.95
4	Дубленка-женская	2500000	10	22.06.95
1	Телевизор "Sony"	2560000	12	01.12.94
10	Холодильник "Наст"	3350000	8	25.01.96

Рис. 3.2. Результат выполнения задания 3.7

После выполнения задания выйдите из СУБД FoxPro в DOS и убедитесь в наличии созданного индексного файла sklad01.idx.

Если индексный файл уже создан во время предыдущего сеанса

работы с СУБД FoxPro, то его нужно открыть при открытии dbf-файла в том случае, если предполагается индексный поиск. Индексные файлы могут быть открыты одновременно с базой данных командой

```
USE [<dbf-файл>] [INDEX <список открываемых индексных файлов>;  
  [ORDER <ВырN>]/[ORDER <idx-файл>] /;  
  [ORDER TAG <имя тега> OF <cdx-файл>]];  
  [ASCENDING/DESCENDING]
```

Опции команды:

INDEX <список открываемых индексных файлов> - перечень открываемых индексных файлов. Первый в списке индексный файл считается главным в том случае, если не использована опция ORDER;

ORDER <ВырN> - указание номера (в списке индексных файлов) главного индексного файла;

ORDER <idx-файл> - главный индексный файл, задаваемый непосредственным указанием его имени;

ORDER TAG <имя тега> OF <cdx-файл> - главный индексный файл, задаваемый именем тега мультииндексного файла;

ASCENDING/DESCENDING - порядок использования индексного файла (по возрастанию/по убыванию). Опция работает вне зависимости от того, с какой опцией этот файл создавался.

В случае, когда база данных уже открыта, открыть индексные файлы можно командой

```
SET INDEX TO [<список открываемых индексных файлов>;  
  [ORDER <ВырN>]/[ORDER <idx-файл>] /;  
  [ORDER TAG <имя тега> OF <cdx-файл>]];  
  [ASCENDING/DESCENDING] [ADDITIVE]
```

Опции команды:

ADDITIVE - открытие новых индексных файлов не закрывает старые.

Остальные опции идентичны соответствующим опциям предыдущей команды.

Закреть все индексные файлы можно следующими командами:

```
SET INDEX TO (без каких-либо параметров)  
CLOSE INDEX
```

ИНДЕКСНЫЙ ПОИСК

Индексный файл не только упорядочивает базу данных, но и

ускоряет поиск в ней по ключу, заданному в индексном файле. Ускоренный поиск можно осуществить командой

```
SEEK <выражение>
```

Команда использует не последовательный, а так называемый двоичный поиск (поиск скачками), что позволяет быстро локализовать номер нужной записи. Команда SEEK разыскивает только одну первую запись, удовлетворяющую условию <выражение>=<поле>. При этом открытый индексный файл должен быть результатом индексирования dbf-файла, осуществленного по этому полю.

Пример. Произведем в базе данных sklad01.dbf индексный поиск записи, соответствующей товару ценой 220 тыс. р., считая, что база данных проиндексирована по полю St1 в результате выполнения задания 3.7.

```
USE sklad01 INDEX sklad01
SET DATE GERMAN
SEEK 220000
CLEAR
DISPLAY Naim, St1, Kol, Dtp
```

Если вдруг в базе данных не окажется записи, поле которой точно соответствует указанному в команде SEEK выражению, то поиск окажется безрезультатным. Такая ситуация может возникнуть, например, в случае, когда пользователь не помнит или не знает точного значения параметра, по которому осуществляется поиск. Перед командой SEEK может быть выполнена команда

```
SET NEAR ON
```

При этом в случае неудачного поиска указатель установится на ближайшую запись, значение анализируемого поля которой максимально совпадает с указанным в команде поиска. Вызвав после этого команду BROWSE и пролистав данные в базе вблизи найденного места, мы получим возможность отобрать подходящие записи. Такой способ поиска данных получил название "мягкого" или приближительного.

ЗАДАНИЕ 3.8

В базе данных sklad01.dbf с использованием соответствующего индексного файла подберите товар (безразлично какой), который можно было бы приобрести на сумму 1 млн р., считая, что в ре-

зультате выполнения задания 3.7 файл sklad01.dbf проиндексирован по полю St1.

Прежде всего необходимо осуществить в командном окне СУБД FoxPro выполнение следующих команд:

```
USE sklad01 INDEX sklad01
SET DATE GERMAN
SET NEAR ON
SEEK 1000000
BROWSE
```

Перемещая курсор вблизи записи, на которую будет установлен указатель записи, подберите требуемый товар.

3.6. Сортировка данных

Физическое упорядочение записей базы данных можно осуществить командой сортировки данных

```
SORT TO <имя нового файла> [ASCENDING/DESCENDING] ON <поле>;
[<границы>] [FOR<условие>] [WHILE<условие>] [FIELDS<поля>]
```

В результате выполнения этой команды создается новый dbf-файл, в котором записи упорядочены по возрастанию (ASCENDING) или убыванию (DESCENDING) какого-либо поля. По умолчанию действует опция ASCENDING. В новом файле будут присутствовать поля, перечисленные в списке полей FIELDS, а также записи, удовлетворяющие FOR- и WHILE-условиям (если они заданы).

Пример.

```
USE sklad01
SORT TO skladst1 ON St1 FIELDS Naim, St1, Kol
```

Создается новый файл skladst1.dbf, имеющий поля Naim, St1 и Kol, в котором записи расположены по возрастанию цены товара.

К достоинствам такого метода упорядочения записей относятся наглядность и простота. Среди основных недостатков - большой объем памяти, требующейся для хранения новой базы данных. На самом деле, в результате сортировки записей создается файл, сопоставимый по размеру с исходным (его размер зависит от числа вошедших в него полей и записей). Причем этот файл не содержит никакой новой информации и, по существу, дублирует исходный. В этом отношении индексирование базы данных предпочтительнее, поскольку размер индексного файла в несколько раз меньше, чем исходного dbf-файла.

3.7. Манипулирование файлами баз данных

Под манипулированием файлами баз данных подразумевается осуществление следующих операций: удаление, переименование, копирование файлов, сортировка записей, копирование структуры файла базы данных и дополнение базы данных записями из другого файла. С некоторыми командами, позволяющими манипулировать файлами, мы уже знакомы. Среди них - команды SORT и ERASE.

Команда

```
COPY FILE <имя исходного файла> TO <имя файла-копии>
```

позволяет копировать любой файл (не обязательно dbf-файл) в новый. Расширение для обоих файлов указывать обязательно.

Для копирования открытого dbf-файла в новый файл можно воспользоваться командой

```
COPY TO <имя нового файла> [<границы>] [FIELDS<поля>];  
[FOR<условие>] [WHILE<условие>] [WITH CDX]
```

В новый файл могут быть скопированы только те поля, которые перечислены в опции FIELDS, и только те записи, которые удовлетворяют FOR- и WHILE-условиям. По умолчанию копируются все поля и все записи базы данных. Копирование можно осуществить вместе с мультииндексным файлом.

СУБД FoxPro предоставляет пользователю возможность осуществить полное или частичное копирование структуры уже существующего открытого файла базы данных командой

```
COPY STRUCTURE TO <имя нового файла> [FIELDS<поля>]
```

При этом можно скопировать во вновь создаваемый пустой dbf-файл только часть структуры открытой базы данных. Опция FIELDS позволяет управлять этим процессом.

Для копирования в активную базу данных информации из других dbf-файлов можно воспользоваться командой

```
APPEND FROM <имя файла-источника> [FIELDS<поля>];  
[FOR<условие>] [WHILE<условие>]
```

Указанная команда позволяет копировать записи, удовлетворяющие FOR- и WHILE-условиям, а также поля, объявленные опцией FIELDS.

Для переименования любого не открытого в данный момент файла используется команда

```
RENAME <старое имя файла> TO <новое имя файла>.
```

Расширение имени указывать обязательно.

3.8. Математическая обработка данных

Команда

```
COUNT [<границы>] [WHILE<условие>] [FOR<условие>];  
      [TO <переменная>]
```

позволяет подсчитать и сохранить в <переменной> число записей в заданных <границах>, удовлетворяющих назначенным <условиям>.

Команда

```
SUM [<границы>] [WHILE<условие>] [FOR<условие>] <поле>;  
      [TO <переменная>]
```

суммирует значения указанного числового <поля> и помещает значение суммы в <переменную>. Суммируются значения поля записей, удовлетворяющих FOR- и WHILE-условиям, а также находящихся в заданных <границах>. Команда допускает указать одновременно несколько суммируемых полей. При этом необходимо задать столько же переменных для полученных сумм.

Команда

```
AVERAGE [<границы>] [WHILE<условие>] [FOR<условие>] <поле>;  
      [TO <переменная>]
```

аналогична по своему действию предыдущей команде, но позволяет вычислять среднее арифметическое поля или нескольких полей.

Пример. SUM FOR St1>1000000 Kol,St1 TO SFINKS, IDOL

Данная команда для записей, имеющих в поле St1 число, превышающее 1000000, находит суммарные значения полей Kol,St1 и помещает обе полученные суммы в переменные SFINKS, IDOL соответственно.

Команда CALCULATE позволяет осуществить эти же и некоторые другие математические операции над полями баз данных.

ЗАДАНИЕ 3.9

Создайте на диске новую базу данных skl01.dbf, содержащую те же поля, что и база данных sklad01.dbf, за исключением Метод-поля. Скопируйте в новую базу данных из файла sklad01.dbf записи, сделанные в 1995 г. Подсчитайте число записей новой базы данных, а также общую сумму ценностей, хранящихся на складе. Получите соответствующую распечатку.

В командном окне необходимо выполнить последовательность команд:


```

USE sklad01
SET TALK OFF
SET DATE GERMAN
COPY STRUCTURE TO skl01.dbf FIELDS Naim, Dtp, Kol, St1, Stm
USE skl01
APPEND FROM sklad01.dbf FIELDS Naim, Dtp, Kol, St1, Stm ;
FOR Dtp<{01.01.96}.AND.Dtp>={01.01.95}
COUNT TO k  && Подсчет в переменной k числа записей
SUM Stm TO s  && Подсчет в переменной s суммы значений поля Stm
SET PRINTER ON
?'                                     ПЕРЕЧЕНЬ'
?'          товаров на складе, поступивших в 1995 г.'
?
?' Наименование          Дата пост. Кол-во   Цена      Стоимость'
SET HEADING OFF
DISPLAY ALL OFF
?'Число записей в БД - ',k
?'Сумма ценностей на складе - ',s,' р.'
?
SET PRINTER OFF
SET TALK ON

```

В этой программе использованы еще неизвестная вам команда вывода ?, дающая возможность выводить в одной строке текстовые константы (в апострофах) и переменные. Эта команда позволяет вывести на принтер заголовки к полям и ко всей базе данных. Команда-установка SET PRINTER ON/OFF предоставляет возможность включить/отключить принтер, а команда SET TALK ON/OFF управляет процессом вывода на экран результатов выполнения многих команд. В частности, выполнение команд SUM, COUNT, AVERAGE и др. сопровождается выводом на экран вспомогательной информации, которая часто только мешает оператору. Команда SET TALK OFF подавляет вывод ненужной информации на экран.

Возможен и другой вариант выполнения этого задания. Вместо команд копирования структуры и дополнения базы данных записями из исходного файла можно воспользоваться командой копирования

```

COPY TO skl01.dbf FIELDS Naim, Dtp, Kol, St1, Stm ;
FOR Dtp<{01.01.96}.AND.Dtp>={01.01.95}

```

ЗАДАНИЕ 3.10

1. Создайте на диске базу данных, содержащую информацию о членах вашей семьи, родственниках, однокурсниках, друзьях или просто знакомых (не менее 15 чел.). Имя файла базы данных задайте по принципу fio.dbf, где fio - первые буквы вашей фамилии, имени, отчества (латинские). Созданная база данных должна иметь следующие поля: фамилия и инициалы, пол, дата рождения, возраст в годах, домашний адрес. Кроме того, она должна содержать Мемо-поле, в котором хранится информация о месте работы и занимаемой должности. Выберите названия полей (латинскими буквами), их тип и размер, создайте структуру dbf-файла и заполните его, как показано на рис. 3.3. Получите распечатку, используя команду DISPLAY или LIST.

2. Используя ту же команду, получите распечатку записей базы данных, хранящих информацию о всех мужчинах (или женщинах), родившихся до (или после) даты рождения кого-либо из лиц, внесенных в вашу базу данных.

3. Проиндексируйте базу данных по полю, хранящему фамилию и инициалы. Получите распечатку, аналогичную полученной при выполнении задания 3.7.

4. Проиндексируйте базу данных по полю, хранящему информацию о дате рождения. Получите соответствующую распечатку.

5. Составьте отчет о проделанной работе, включающий в себя результаты выполнения заданий 1.1-3.10, а именно: тексты самих заданий, тексты написанных программ с необходимыми пояснениями и полученные распечатки.

Fio	Pol	Drd	Wsr	Adr	Ргтл
Иваненко П.Н.	М	02.12.35	61	Воинова 14-113	Мемо
Большакова С.В.	Ж	23.01.72	24	Веселовского 68-12	Мемо
Краилин В.Ф.	М	07.06.39	57	Сушинского 12а-115	Мемо
Липатова Н.А.	Ж	14.06.78	19	Сушинского 4-15	Мемо
Краилина В.Г.	Ж	16.05.47	49	Сушинского 12а-115	Мемо
Краилина К.Г.	Ж	14.06.42	54	Сушинского 12а-115	Мемо
Краилин С.В.	М	18.09.64	32	пр.70-летия Окт. 18-1	Мемо
Кузоваткина В.В.	Ж	13.12.26	70	Расковой 45-34	Мемо

Рис. 3.3. Пример заполнения базы данных

ЗАКЛЮЧЕНИЕ

В данной работе рассмотрены некоторые основные команды СУБД FoxPro. Их практическое освоение будет способствовать получению студентами начальных навыков работы с базами данных, покажет им возможности системы FoxPro по обработке значительных объемов информации. К сожалению, в небольшой работе невозможно отразить все важные команды, в частности, позволяющие организовать удобный интерфейс при создании систем обработки данных, генерировать выходные документы (отчеты, сводки информации и т.п.), обеспечивать более сложные операции по управлению данными. Тем не менее авторы надеются, что основная задача ими выполнена: читатель получил необходимый минимум знаний, опираясь на который можно будет продолжить совершенствование своего умения осуществлять власть над информацией.

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. Попов А.А. Программирование в среде СУБД FoxPro 2.0. Построение систем обработки данных. М.: Радио и связь, 1994. 354 с.
2. Фигурнов В.Э. IBM PC для пользователя. 6-е изд., перераб. и доп. М.: ИНФРА-М, 1995. 432 с.
3. Масловский Е.К. Англо-русский словарь пользователя персональной ЭВМ / Моск. междунар. шк. переводчиков. М., 1992. 223 с.

СОДЕРЖАНИЕ

Предисловие	3
1. Общие технические характеристики СУБД FoxPro	4
1.1. Структура данных в СУБД FoxPro	4
1.2. Синтаксис команд СУБД FoxPro	6
2. Создание файла базы данных	9
2.1. Создание структуры файла базы данных	9
2.2. Заполнение базы данных	12
3. Управление базами данных	15
3.1. Редактирование баз данных	15
3.2. Перемещения в базе данных	20
3.3. Просмотр и изменение в базе данных	20
3.4. Локализация и поиск данных в базе	25
3.5. Индексирование баз данных	26
3.6. Сортировка данных	31
3.7. Манипулирование файлами баз данных	32
3.8. Математическая обработка данных	33
Заключение	36
Список рекомендуемой литературы	37

ОСНОВЫ ПРОГРАММИРОВАНИЯ В СРЕДЕ СУБД FOXPRO
Методические указания

Составители: БАЙНЕВ Валерий Федорович
СИДОРОВ Андрей Александрович

Редактор Е. С. Руськина
Технический редактор Л. Н. Чебакова

Лицензия ЛР № 040696 от 04.01.95. Сдано
в набор 09.01.97. Подписано в печать 17.01.97.

Формат 60x84 1/16. Бумага офсетная.

Печать офсетная. Усл. печ. л. 2,32.

Усл. кр.-отт. 2,44. Уч.-изд. л. 2,10.

Тираж 200 экз. Заказ №717 .

Саранский кооперативный институт МУПК
430027, г. Саранск, ул. Транспортная, 17.

Типография "Рузаевский печатник"
Министерства печати и информации РМ.
431460 г. Рузаевка, ул. Трынова, 67а.