

Лабораторная работа № 6

Тема : Написание программного файла и работа с несколькими табличными файлами.

Цель работы: 1. Написать и сохранить на диске программный файл.

2. Объединить несколько табличных файлов в одну базу данных.

Литература:

1. Омельченко Л. Д. Шевякова, Visual FoxPro 9.0 Санкт-Петербург, 2009
2. Т.В.Мухина, Visual FoxPro 9.0 Санкт-Петербург, 2009
3. В.Клепинин, Т.Агафонова, Visual FoxPro 9.0 Санкт-Петербург, 2008
4. Голицына О.Л., Максимов Н.В., Попов И.И. Базы данных Учебное пособие. М.Форум-Инфра-М,2009
5. Фуфаев Э.В., Фуфаев Д.Э. Базы данных. Учебное пособие. М. Издат.центр «Академия», 2006

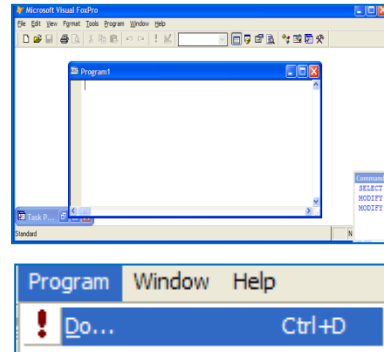
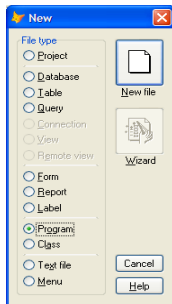
Теоретические сведения

Для создания программного файла используется либо встроенный текстовый редактор, либо любой текстовый редактор. Программный файл пишется с помощью команд FoxPro как обычная программа.

При работе со встроенным текстовым редактором работают все стандартные приемы набора и редактирования текста. Для запуска встроенного редактора надо из главного меню подать команду:

File → **New** → из появившегося на экране списка выбрать тип файла **Program** и нажать кнопку **New File**.

в рабочем окне встроенного редактора набрать текст программы. Для запуска программы надо в графическом меню нажать кнопку



Для вновь созданного программного файла FoxPro потребует присвоить ему имя и выведет на экран соответствующую диалоговую панель. После сохранения на диске программный файл **автоматически будет поставлен** на выполнение. При обнаружении ошибок в тексте программы на экран выводятся соответствующие сообщения.

При объединении нескольких табличных файлов надо обеспечить выполнение следующих условий (подробнее см. лабораторную работу №5) :

- **построить индексы по общим ключевым полям;**
- **открыть табличные файлы вместе со своими индексными файлами, причем каждый табличный файл в своей рабочей области;**
- **объединяемые табличные файлы должны иметь общее поле (хотя бы попарно).**

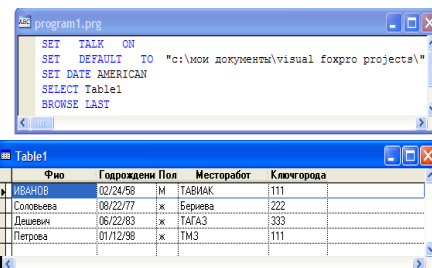
Вопросы для самопроверки.

1. Укажите назначение программного файла.
2. Как программный файл поставить на выполнение?
3. Укажите приемы редактирования программного файла.
4. Какой командой задается активная рабочая область?
5. Укажите назначение префикса.

Задание часть 1

1. Заполнить таблицы **GOROD** и **ULICA** для всех записей основной таблицы
2. Проиндексировать каждую из этих таблиц
3. Создать программный файл по открытию всех таблиц вместе со своими индексными файлами.

- Установить рабочий каталог по умолчанию
- Установить тип даты - **BRITISH**
- Каждую таблицу открыть в своей рабочей области с присвоением алиаса.
- Предусмотреть: отображение на экране результатов выполнения команд
- Организовать поиск в основной таблице по индексному полю
- Установить курсор на 1-ю запись и вывести на экран 3 последующие записи



5. Написать программный файл по установлению взаимосвязей между таблицами и выводу на экран данных из взаимосвязанных таблиц.

Задание часть 2

1. Открыть новый проект - **PROBA**
2. Создать программный файл – **PROG2**, в котором предусмотреть комментарии к каждой строке. Например, **CLOSE ALL && закрыть всё**
3. Закрыть ранее открытые файлы
4. Установить путь к своему рабочему каталогу
5. Установить тип даты - **GERMAN**
6. Создать новую базу данных - **BAZA**
7. В базе данных создать таблицу **TABLE1** с полями

Фамилия	символьное	20
Имя	символьное	10
Дата рождения	дата	8
Отчество	символьное	20

Например:

CREATE TABLE table1 (fam c(20), name c(10), otchestv c(20), datarogd d(8))

8. Заполнить поля таблицы – команда добавления пустой записи (для 3-х записей)

9. Просмотреть содержимое таблицы **TABLE1**
 10. Создать ещё одну таблицу – **TABL2** с полями:

Фамилия	символьное	20
Имя	символьное	10
Отчество	символьное	20
Дата рождения	дата	8
Предмет 1	символьное	20
Оценка 1	числовое	1
Предмет 2	символьное	20
Оценка 2	числовое	1
Предмет 3	символьное	20
Оценка 3	числовое	1

11. Заполнить поля таблицы – команда добавления пустой записи (для 3-х записей)
 12. Просмотреть содержимое таблицы **TABL2**
 13. Выбрать рабочую область 1
 14. Активизировать в выбранной области таблицу **TABLE1**
 15. Индексировать таблицу **TABLE1** - создать тег **FAM1** по полю - фамилия

16. Индексировать таблицу **TABLE1** - создать тег **DAT1** по полю – дата рождения
 17. Выбирать рабочую область 2
 18. Активизировать в выбранной области таблицу **TABLE2**
 19. Индексировать таблицу **TABLE2** - создать тег **FAM2** по полю - фамилия
 20. Установить связь «один-к-одному» по полю – фамилия с таблицей **TABLE1**
 21. Осуществить поиск в таблице **TABLE1** по полю – имя
 22. Установить курсор на 2-ю запись таблицы **TABLE1** и вывести на экран 2-ю и 3-ю записи
 23. Закрыть все таблицы и базы
 24. Оформить отчет.

Образец программного файла:

SET DEFAULT TO "F:\MyProject" && установка каталога для работы
CREATE PROJECT proj1 NOWAIT SAVE && создание проекта. **NOWAIT SAVE** означает, что окно проекта не будет закрыто при переходе на следующую строку программы
CREATE && создание таблицы
APPEND mod && включение мода изменения таблицы
BROWSE && предьявление таблицы на экран

MODIFY STRUCTURE && изменение структуры таблицы
CREATE DATABASE database1 && создание базы данных
MODIFY PROJECT proj1 && изменение структуры проекта
 && добавление базы данных в проект
 && добавление таблиц в БД

USE database1! table1 && загрузка в ОП таблицы
INDEX on pole1 TAG tg1 ASCENDING && создание тега с сортировкой по возрастанию
INDEX on pole2 TAG tg2 DESCENDING && создание тега с сортировкой по убыванию

LOCATE FOR pole1="1" && последовательный поиск методом перебора
BROWSE && предьявление результатов поиска на экран. Установка курсора на найденное значение
CONTINUE && продолжение поиска
BROWSE && предьявление результатов поиска на экран. Установка курсора на найденное значение

USE database1!table1 && загрузка в ОП таблицы
SET ORDER TO tg1 && установка основного тега
SEEK "1" && быстрый поиск по индексному полю

Работа в окне Command

SELECT 1
BROWSE
SELECT database1!table2
SELECT table2
USE table2 AGAIN IN 0
SELECT Table2
BROWSE LAST
SELECT Table2
SET RELATION TO pole1 INTO table1
USE table1 AGAIN IN 0
SELECT Table1
BROWSE LAST
SELECT Table2
BROWSE LAST
SET RELATION TO pole1 INTO table1
SELECT Table1
SET ORDER TO TG1 && POLE1
SELECT Table2

SET RELATION TO pole1 INTO table1
SELECT Table1
SELECT Table2
SELECT Table1
SET ORDER TO TG1 && POLE1
SELECT Table2
SET RELATION TO pole1 INTO table1
MODIFY DATABASE
SET SKIP TO table1, table2
SELECT Table1
SET SKIP TO table1, table2
SET RELATION TO pole1 INTO table1
SELECT Table2
SET RELATION TO pole1 INTO table1