

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

Государственное образовательное учреждение
высшего профессионального образования
«Оренбургский государственный университет»

Колледж электроники и бизнеса

Кафедра вычислительной техники и математики

Т.И. ИШТЕРЯКОВА

БАЗЫ ДАННЫХ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ
ЛАБОРАТОРНЫХ РАБОТ

Рекомендовано к изданию Редакционно-издательским советом
государственного образовательного учреждения
высшего профессионального образования
«Оренбургский государственный университет»

Оренбург 2009

УДК 621.38(075.32)

ББК 32.81 я72

И 97

Рецензент

Заместитель директора по НМР Кузюшин С.А.

Иштерякова Т.И.

Базы данных [Текст]: Методические указания к лабораторным работам /Т.И.Иштерякова – Оренбург: ГОУ ОГУ, 2009. – 69 с.

И 97

Методические указания предназначены для проведения лабораторных работ, обеспечивающих учебный процесс по дисциплине “Базы данных” в колледже Электроники и бизнеса ОГУ для студентов 3 курса в 6 семестре специальности 230105 Программное обеспечение вычислительной техники и автоматизированных систем очной формы обучения, а также для студентов заочной формы обучения.

Методические указания к лабораторным работам составлены с учетом Государственного образовательного стандарта среднего профессионального образования по направлению подготовки дипломированных специалистов - утвержденного 30.12.2003 Министерством Образования Российской Федерации.

ББК 32.81 я72

©Иштерякова Т.И.

©ГОУ ОГУ, 2009

Содержание

Введение	6
1 Проектирование структуры базы данных	7
1.1 Цель работы	7
1.2 Постановка задачи	7
1.3 Теоретическая часть	7
1.3.1 Основные понятия	7
1.3.2 Нормализация отношений	8
1.3.3 Инфологическое и даталогическое проектирование	8
1.4 Практическая часть	10
1.4.1 Варианты заданий	10
1.4.2 Порядок выполнения работы	11
1.4.3 Содержание отчета	11
1.4.4 Вопросы для защиты работы	11
2 Работа с таблицами, записями и базами данных	12
2.1 Цель работы	12
2.2 Постановка задачи	12
2.3 Теоретическая часть	12
2.3.1 Команды установки	12
2.3.2 Основные команды для работы с таблицами	13
2.3.3 Команды редактирования записей	14
2.3.4 Работа с базами данных	16
2.3.5 Установление взаимосвязи с помощью главного меню	17
2.4 Практическая часть	20
2.4.1 Варианты заданий	20
2.4.2 Порядок выполнения работы	22
2.4.3 Содержание отчета	22
2.4.4 Вопросы для защиты работы	23
3 Управление базами данных	23
3.1 Цель работы	23
3.2 Постановка задачи	23
3.3 Теоретическая часть	24
3.3.1 Сортировка данных	24
3.3.2 Поиск данных	25
3.3.3 Фильтрация данных	27
3.4 Практическая часть	27
3.4.1 Варианты заданий	27
3.4.2 Порядок выполнения работы	29
3.4.3 Содержание отчета	30
3.4.4 Вопросы для защиты работы	30
4 Работа с программным файлом	30
4.1 Цель работы	30
4.2 Постановка задачи	30

4.3 Теоретическая часть	31
4.3.1 Создание программного файла	31
4.3.2 Открытие и запуск программного файла	32
4.4 Практическая часть	32
4.4.1 Варианты заданий	32
4.4.2 Порядок выполнения работы	33
4.4.3 Содержание отчета	33
4.4.4 Вопросы для защиты работы	33
5 Создание меню с помощью конструктора	34
5.1 Цель работы	34
5.2 Постановка задачи	34
5.3 Теоретическая часть	34
5.3.1 Действия, необходимые для создания меню	34
5.3.2 Создание меню в виде строки	35
5.3.3 Сохранение, генерация и запуск меню	43
5.3.4 Создание всплывающего меню	44
5.4 Практическая часть	45
5.4.1 Порядок выполнения работы	45
5.4.2 Содержание отчета	45
5.4.3 Вопросы для защиты работы	45
6 Организация интерфейса пользователя и создание отчетов	46
6.1 Цель работы	46
6.2 Постановка задачи	45
6.3 Теоретическая часть	46
6.3.1 Создание экранной формы с помощью Мастера форм — Form Wizard	46
6.3.2 Создание экранной формы с помощью Конструктора форм Form Designer	47
6.3.3 Создание отчетов с помощью мастера	49
6.3.4 Создание отчетов с помощью конструктора	55
6.4 Практическая часть	56
6.4.1 Порядок выполнения работы	56
6.4.2 Содержание отчета	56
6.4.3 Вопросы для защиты работы	57
7 Создание триггеров и достоверность данных	57
7.1 Цель работы	57
7.2 Постановка задачи	57
7.3 Теоретическая часть	57
7.3.1 Условия достоверности ввода данных на уровне записей	57
7.3.2 Триггеры	58
7.4 Практическая часть	60
7.4.1 Порядок выполнения работы	60
7.4.2 Содержание отчета	60
7.4.3 Вопросы для защиты работы	60
8 Запросы к базе данных	61
8.1 Цель работы	61
8.2 Постановка задачи	61

8.3 Теоретическая часть	61
8.3.1 Запросы к базе данных.....	61
8.3.2 Конструктор запросов.....	62
8.3.3 Команды, используемые при формировании запросов.....	64
8.3.4 Сохранение запроса.....	65
8.3.5 Запуск запроса на выполнение.....	66
8.4 Практическая часть	66
8.4.1 Порядок выполнения работы	66
8.4.2 Содержание отчета.....	66
8.4.3 Вопросы для защиты работы	67
Список использованных источников	68

Введение

Создание баз данных после появления реляционного подхода превратилось из искусства в науку, но как показала практика последних лет, все же окончательно его не исключившая. Тем не менее, сейчас это вполне сложившаяся дисциплина, основанная на достаточно формализованных подходах и включающая широкий спектр приемов и методов создания баз данных.

Несмотря на появление новых систем управления базами данных, Visual FoxPro все еще остается одной из наиболее популярных программ в этой области. Это прежде всего связано с тем, что при создании новых версий Microsoft сохраняет совместимость с предыдущими версиями.

Visual FoxPro – это объектно-ориентированный, визуально программируемый язык, управляемый по событиям, который в полной мере соответствует новым требованиям, предъявляемым к современным средствам проектирования.

В данных методических рекомендациях описываются все основные аспекты по работе с базами данных. Представлены все возможные способы выполнения действий по работе с базами данных. Методические указания по каждой лабораторной работе состоят из теоретической и практической частей, что наилучшим образом демонстрирует тесную связь теории и практики.

1 Проектирование структуры базы данных

1.1 Цель работы

Изучить этапы проектирования базы данных. Научится выполнять нормализацию таблиц реляционной базы данных. Научится строить логическую структуру базы данных.

1.2 Постановка задачи

В предметной области, согласно своему варианту, выделить объекты и их свойства. Привести все объекты к 3 нормальной форме. Построить инфологическую и логическую модели базы данных.

1.3 Теоретическая часть

1.3.1 Основные понятия

Объект — это нечто существующее и различимое, обладающее набором свойств.

Сущность — отображение объекта в памяти человека или компьютера.

Параметр — конкретное значение любого из свойств объекта.

Атрибут — конкретное значение любого из свойств сущности.

Запись — это одна строка таблицы (или нескольких таблиц), полностью описывающая одну сущность. Каждая запись состоит из конечного числа полей.

Поле — это один элемент записи, в котором хранится конкретное значение атрибута.

Первичный ключ — это атрибут или группа атрибутов, которые однозначно определяют каждую запись в таблице.

Связь — это функциональная зависимость между сущностями.

Нормализация отношений — это процесс построения оптимальной структуры таблиц и связей в реляционной БД.

Концептуальная модель данных — это совокупность концептуальных требований, выдвинутых сотрудниками структурных подразделений фирмы.

В результате отображения концептуальной модели на СУБД будет получена *логическая модель данных*.

Отображение логической модели на конкретные технические средства называется *физической моделью данных*.

1.3.2 Нормализация отношений

Для таблицы будут выполнены условия первой нормальной формы, если:

- каждое поле (концептуальное требование) неделимо;
- отсутствуют повторяющиеся поля или группы полей.

Условия второй нормальной формы:

- выполняются условия первой нормальной формы;
- первичный ключ однозначно определяет всю запись;
- все поля зависят от первичного ключа;
- первичный ключ не должен быть избыточным.

Условия третьей нормальной формы:

- выполняются условия второй нормальной формы;
- каждое не ключевое поле не должно зависеть от другого не ключевого поля.

1.3.3 Инфологическое и даталогическое проектирование

Инфологическая модель представляется в виде ER-диаграммы и описания. Например, в предметной области «Воспитательная работа в колледже» можно выделить объекты: КУРАТОР, ГРУППА и СТУДЕНТ. Связь между этими объектами будет 1:М и 1:1. ER-диаграмма для данного примера представлена на рисунке 1.

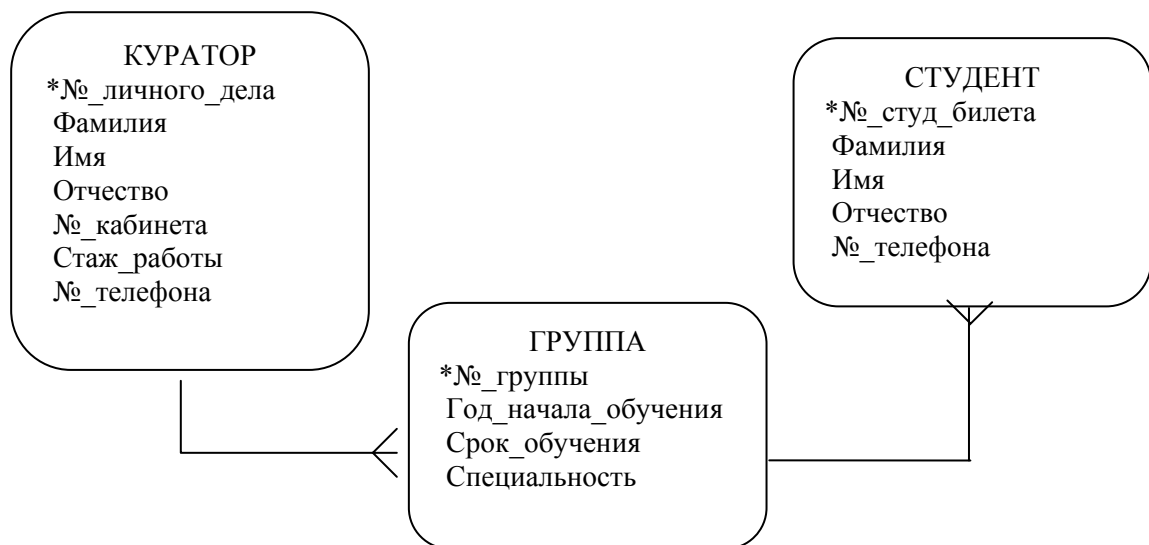


Рисунок 1 – Пример ER-диаграммы

Для каждой связи в ER-диаграмме составляются описания. Для данного примера можно составить следующие описательные предложения:

- 1) каждый куратор должен курировать одну и более групп;

- 2) за каждой группой должен быть закреплен только один куратор;
- 3) в каждой группе должен обучаться один и более студентов;
- 4) каждый студент должен обучаться только в одной группе.

Даталогическая модель базы данных представляется в виде таблиц (см. таблицу 1).

Таблица 1 – Структура реляционной таблицы

Имя файла						
Поле		Признак ключа	Формат поля			
Имя (обозначение)	Полное наименование		Тип	Длина	Точность (для чисел)	N/NN
имя 1						
...						
имя n						

В столбце Признак ключа можно указать:

- PK (primary key) – первичный ключ;
- UK (unique key) – уникальный ключ;
- FK (foreign key) – внешний ключ.

В столбце N/NN – указывается может ли поле быть пустым:

- N (null) – поле может быть пустым;
- NN (not null) – поле не может быть пустым.

Для нашего примера даталогическая модель объекта ГРУППА представлена в таблице 2.

Таблица 2 – Структура объекта ГРУППА

Group dbf						
Поле		Признак ключа	Формат поля			
Имя (обозначение)	Полное наименование		Тип	Длина	Точность	N/NN
Number_gr	№ группа	PK	Character	5		NN
Year_stady	Год начала обучения		Date	10		NN
Time_stady	Срок обучения		Character	25		NN
Specialn	Специальность		Character	20		NN
Number_pr	№ личного дела преподавателя	FK	Integer	10	-	NN

1.4 Практическая часть

1.4.1 Варианты заданий

1) Предметная область: "Авиакомпания". Концептуальные требования: номер самолета, модель самолета, технические характеристики самолета, № рейса, пункт отправления, пункт прибытия, время полета, № билета, ФИО пассажира, цена билета;

2) Предметная область: "Компьютерные лаборатории". Концептуальные требования: номер кабинета, название кабинета, количество машин в классе, серийный номер компьютера, технические характеристики компьютера, личный номер ответственного за класс, ФИО ответственного, телефон ответственного;

3) Предметная область: «Детский сад». Концептуальные требования: ФИО воспитателя, образование воспитателя, ФИО ребенка, дата рождения ребенка, № группы, название группы, № свидетельства о рождении ребенка;

4) Предметная область: "Бюро путешествий". Концептуальные требования: № лицензии агентства отдыха, телефон, адрес, код путевки, срок действия, дата начала действия, место отдыха, форма отдыха, ФИО клиента, № паспорта клиента, цена путевки, телефон клиента;

5) Предметная область: "Искусство". Концептуальные требования: название работы, цена работы, ФИО автора, телефон автора, название музея, телефон музея, адрес музея, дата выставки, название выставки;

6) Предметная область "Курсы". Концептуальные требования: № лицензии, наименование курсов, продолжительность обучения, дата начала, ФИО преподавателя, ФИО слушателя, паспорт слушателя, телефон слушателя;

7) Предметная область: "Учебный процесс". Концептуальные требования: личный номер студента, ФИО студента, дата рождения, группа, код дисциплины, название дисциплины, количество часов, код преподавателя, ФИО преподавателя;

8) Предметная область "Товар". Концептуальные требования: код производителя, название производителя, адрес производителя, код товара, характеристика товара, дата выпуска, № магазина, название магазина, адрес магазина, телефон магазина;

9) Предметная область: "Производство деталей". Концептуальные требования: код детали, наименование детали, размер, характеристика, № исполнителя, ФИО исполнителя, № договора, дата подписания, количество деталей, срок исполнения;

10) Предметная область: "Автомобиль". Концептуальные требования: № автомобиля, марка автомобиля, дата выпуска, технические характеристики, цвет, паспорт владельца, ФИО владельца, адрес владельца, № тех. осмотра, дата;

11) Предметная область: "Спортивный клуб". Концептуальные требования: название клуба, адрес, телефон, ФИО тренера, телефон тренера, ФИО посетителя, телефон посетителя;

12) Предметная область: "Соревнования". Концептуальные требования: название команды, количество игроков, ФИО тренера, дата соревнований, исход соревнований

1.4.2 Порядок выполнения работы

- 1) описать предметную область, согласно своему варианту;
- 2) выделить объекты с их параметрами;
- 3) привести к третьей нормальной форме все сущности;
- 4) построить ER-диаграмму, составить описательные предложения, соответствующие ER-диаграмме;
- 5) построить на основе ER-диаграммы логическую структуру базы данных в виде таблиц;
- 6) сделать вывод о проделанной работе;
- 7) оформить отчет.

1.4.3 Содержание отчета

- 1) тема, цель;
- 2) постановка задачи, вариант задания;
- 3) нормализация отношений;
- 4) ER-диаграмма и описание;
- 5) структура таблиц базы данных;
- 6) вывод.

1.4.4 Вопросы для защиты работы

- 1) Дайте определения основным понятиям теории баз данных.
- 2) Назовите условия 1,2 и 3 нормальной формы.
- 3) Какие типы взаимосвязей существуют?
- 3) Что будет результатом инфологического проектирования базы данных?
- 4) По каким правилам строится ER-диаграмма?
- 5) Назовите типы ключевых полей.
- 6) Как составляется даталогическая модель объектов?
- 7) Назовите все этапы проектирования базы данных.
- 8) Назовите требования, предъявляемые к базе данных.

2 Работа с таблицами, записями и базами данных

2.1 Цель работы

Получить элементарные навыки работы в СУБД Visual Fox Pro. Закрепить и применить на практике теоретические знания по работе с таблицами, записями и базами данных в СУБД Visual Fox Pro.

2.2 Постановка задачи

Создать файл базы данных, согласно своему варианту, и добавить в него табличные файлы, установить взаимосвязи между таблицами.

2.3 Теоретическая часть

2.3.1 Команды установки

Для вывода статус-строки на экран надо в окне Command подать команду *SET STATUS ON*.

В Fox Pro предусмотрены несколько типов дат, которые устанавливаются с помощью команды *SET DATE<тип даты>*. Аргумент *<тип даты>* может содержать одну из констант:

- *AMERICAN* – американский тип даты мм/дд/гг;
- *ANCY* – тип даты *ANCY* гг.мм.дд;
- *BRITISH* – английский тип даты дд/мм/гг;
- *GERMAN* – немецкий тип даты дд.мм.гг;
- *ITALIAN* – итальянский тип даты дд-мм-гг;

Если необходимо установить написание года четырьмя цифрами, то надо подать команду *SET CENTURY ON*.

Команда установки текущего каталога *SET DEFAULT TO <полное имя каталога>*.

Команда задания альтернативных путей поиска *SET PATH TO [<список путей поиска>]*.

Команда управления курсором *SET CONFIRM ON*. Команда включает автоматический перевод курсора в следующее поле после нажатия клавиши ENTER.

Команда подачи звукового сигнала *SET BELL ON*.

Управление выдачей системных сообщений *SET TALK ON|OFF*. Если установлена опция ON, то на экран выводятся системные сообщения о выполнении некоторых команд и многих функций Fox Pro.

2.3.2 Основные команды для работы с таблицами

Табличный файл можно создать следующими способами:

- 1) с помощью команды *CREATE*;
- 2) с помощью Главного меню (File / New);
- 3) с помощью SQL- оператора

Сокращенный формат SQL-оператора для создания таблицы имеет вид:

CREATE TABLE <имя таблице> (<имя поля1> <тип поля1> (<размер поля1>) [, <имя поляN> <тип поляN> (<размер поляN>),...])

Диалоговая панель Table Designer (рисунок 2) служит для описания структуры таблицы. Каждое поле структуры таблицы описывается одной строкой и для него заполняются четыре параметра (характеристики). Назначение параметров поля:

- Name – имя поля. Пишется буквами латинского алфавита, первый символ – обязательно буква, далее можно использовать цифры и знак подчеркивания;
- Type – тип поля. Указывается один из допустимых типов поля либо с клавиатуры, либо открывается раскрывающийся список и на экран выводится список допустимых типов полей, из которого выбирается нужный тип;
- Width – указывается желаемый размер поля, но не больше допустимого.
- Decimal – количество разрядов дробной части числа. Поле доступно только при использовании типа Numeric;
- Null – запрет пустого (нулевого) значения поля.

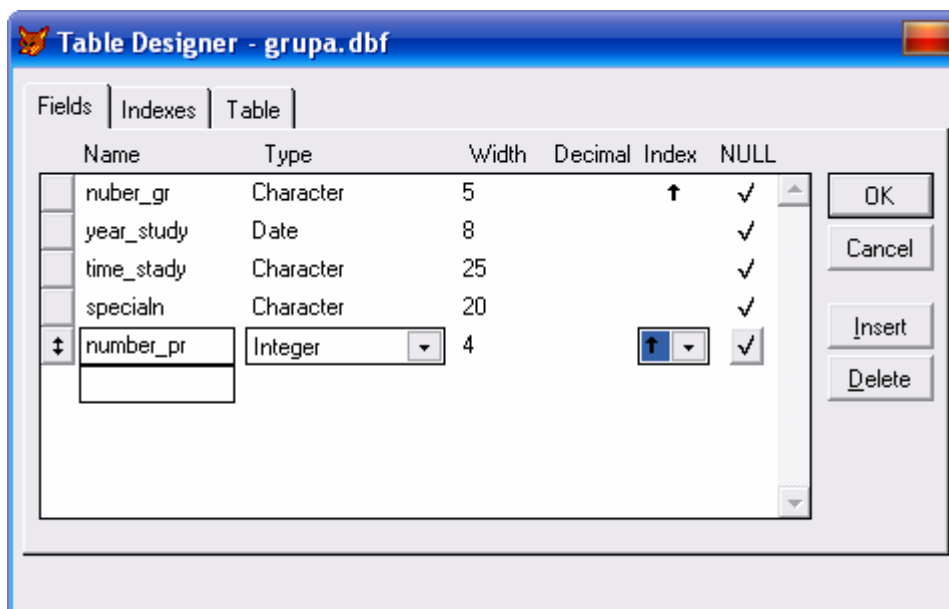


Рисунок 2 – Диалоговая панель Table Designer

Данные в шаблон заносятся по правилам текстовых редакторов. По окончании ввода всех данных (всех записей), их надо сохранить на диске с

помощью одной из команд: *Ctrl + End* или *Ctrl + W*. Если данные сохранять не надо, то нажимают комбинацию клавиш *Ctrl + Q* или клавишу *Esc*.

Модифицировать можно текущую таблицу, при этом из существующей структуры можно удалить ненужное поле (поля), добавить в структуру новое поле (поля), изменить характеристики существующих полей. Из окна *Command* надо подать команду *MODIFY STRUCTURE*.

Открыть ранее созданную таблицу можно двумя способами: с помощью команд (*USE <имя_таблицы>*) или с помощью главного меню (*File/Open*).

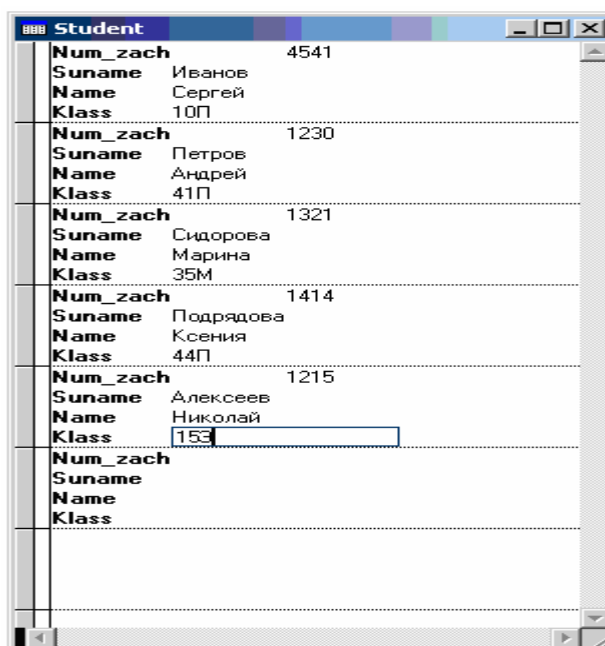
2.3.3 Команды редактирования записей

При работе с базой данных часто возникает необходимость исправить данные в таблицах, то есть выполнить редактирование данных. Предварительно открывается нужный табличный файл. Вывести на экран таблицу можно в двух режимах: режим *Browse* (рисунок 3) и режим *Edit* (рисунок 4).



	Num_zach	Suname	Name	Klass
	4541	Иванов	Сергей	10П
	1230	Петров	Андрей	41П
	1321	Сидорова	Марина	35М
	1414	Подрядова	Ксения	44П
	1215	Алексеев	Николай	153

Рисунок 3 - Режим *Browse*



Num_zach	4541
Suname	Иванов
Name	Сергей
Klass	10П
Num_zach	1230
Suname	Петров
Name	Андрей
Klass	41П
Num_zach	1321
Suname	Сидорова
Name	Марина
Klass	35М
Num_zach	1414
Suname	Подрядова
Name	Ксения
Klass	44П
Num_zach	1215
Suname	Алексеев
Name	Николай
Klass	153
Num_zach	
Suname	
Name	
Klass	

Рисунок 4 - Режим *Edit*

В режиме Browse в окне BROWSE выводится несколько записей. Количество записей определяется размером окна BROWSE, при этом возможна прокрутка как по горизонтали, так и по вертикали. В режиме Edit в окне BROWSE выводится одна запись, при этом записи можно пролистывать вперед (клавиша PgDn) и назад (клавиша PgUp).

Установить режимы Browse и Edit можно несколькими способами:

1) с помощью команд: для задания режима Browse надо в окне Command подать команду *BROWSE*, для задания режима Edit надо в окне Command подать команду *EDIT* или *CHANGE*;

2) с помощью Главного меню: для задания режима Browse надо подать команду View/ Browse, для задания режима Edit надо подать команду View/Edit.

В FoxPro новая запись добавляется в конец таблицы. Можно добавлять как одну, так и несколько записей. Предварительно открывают табличный файл, а затем одним из способов подают команду добавления записи.

1) с помощью команд

в окне Command подают команду *APPEND*.

2) с помощью Главного меню

из Главного меню подают одну из команд:

Table -> Append New Record — для добавления одной записи.

Table -> Append Mode — для добавления нескольких записей.

При подаче любой из команд на экран выводится окно BROWSE в режиме Edit, которое содержит пустой шаблон записи.

Пометить запись к удалению можно следующими способами:

1) с помощью мыши;

2) с помощью клавиатуры (Ctrl + T);

3) с помощью главного меню (Table/Toggle Deletion Mark);

4) программно *DELETE [FOR <L>] [WHILE <L>] [NOOPTIMIZE]*.

Команда *DELETE* без опций помечает к удалению одну текущую запись (рисунок 5).

Num_zach	Surname	Name	Class
4541	Иванов	Сергей	10П
1230	Петров	Андрей	41П
1321	Сидорова	Марина	35М
1414	Подрядова	Ксения	44П
1215	Алексеев	Николай	153

Рисунок 5 - Записи, помеченные к удалению


Физическое удаление помеченных записей можно выполнить двумя способами:

1) с помощью команды *PACK*;

2) с помощью Главного меню (Table/ Remove Deleted Records).

2.3.4 Работа с базами данных

Создать файл базы данных можно несколькими способами:

- 1) с помощью команды *CREATE DATABASE* [<имя базы данных> | ?];
- 2) с помощью Главного меню (File/New);
- 3) с помощью графического меню  .

Файл базы данных имеет расширение .DBC.

Для открытия базы данных надо из Главного меню подать команду File/Open или в графическом меню нажать кнопку Open, на экран выведется диалоговая панель Open, где надо указать имя базы данных и место ее хранения. В результате работы команды Open на экран выведется диалоговая панель Database Designer.

Для того чтобы поместить в открытую базу данных ранее созданную свободную таблицу, надо выполнить одно из действий:

- либо из Главного меню подать команду Database /Add Table;
- либо установить курсор мыши на рабочее поле диалоговой панели

Database Designer и правой кнопкой мыши вызвать на экран контекстное меню, из которого выбрать команду Add Table (рисунок 6).

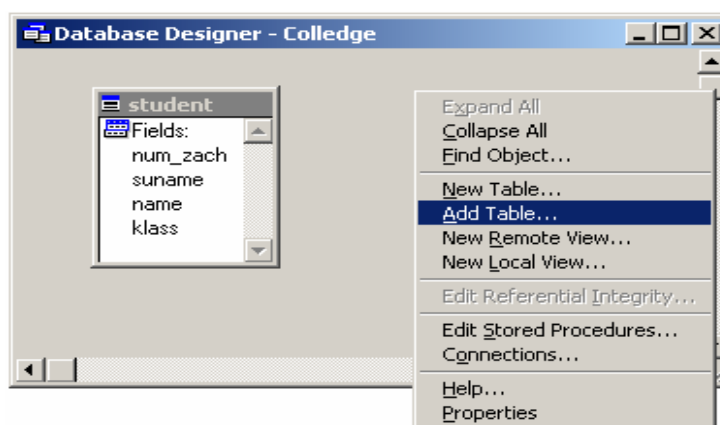


Рисунок 6 - Добавление таблицы в базу данных

Добавить свободную таблицу в открытую базу данных можно также с помощью команды *ADD TABLE* <имя таблицы> | ?

Для освобождения таблицы существуют две команды.

Команда *REMOVE TABLE* <имя таблицы> ? [*DELETE*] удаляет таблицу из базы данных и делает указанную таблицу свободной, оставляя ее в текущем каталоге. Если указана опция *DELETE*, то таблица удаляется не только из базы данных, но и с диска.

Команда *FREE TABLE* <имя таблицы> используется при закрытой базе данных для доступа к указанной таблице. Если опция <имя таблицы> опущена, то на экран выводится диалоговая панель Free Table, которая аналогична диалоговой панели Open, где можно указать имя освобождаемой таблицы.

Освободить таблицу можно через Главное меню командой Database/Remove.

После выбора команды *Remove* надо уточнить свое действие: освободить таблицу (Remove) или удалить таблицу (Delete) нажатием соответствующей кнопки (рисунок 7).

После нажатия одной из кнопок (Remove или Delete) надо подтвердить (или отменить) выбранное действие нажатием кнопки Yes (или No) в появившемся на экране новом запросе.

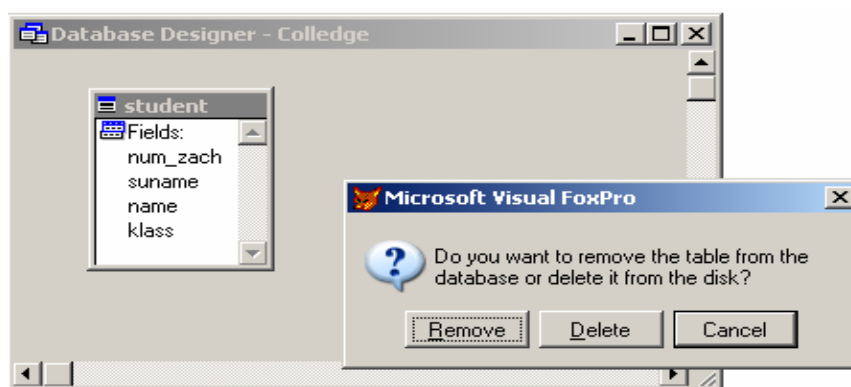


Рисунок 7 - Освобождение (или удаление) таблицы

2.3.5 Установление взаимосвязи с помощью главного меню

Если имеется реляционная база данных, то установить взаимосвязи между таблицами можно с помощью команд главного меню. Для этого предварительно в каждой таблице строится первичный ключ и внешние ключи. Затем выводят на экран диалоговую панель Table Designer. Потом курсор мыши размещают на имени первичного ключа родительской таблицы и буксируют его внутрь дочерней таблицы, устанавливая на имя соответствующего внешнего ключа. Во время буксировки курсор мыши дважды меняет свою форму. По окончании буксировки на экран выводится диалоговая панель Edit Relationship, где надо проверить, а при необходимости уточнить параметры взаимосвязи.

После нажатия кнопки Ok на экран выводится диалоговая панель Database Designer, между именами соответствующих индексов автоматически прорисовывается прямая линия. Со стороны «один» линия начинается с символа «+», а со стороны «много» — с символа «->» (рисунок 8) . Обратная буксировка (от дочерней таблицы к родительской) не допустима.

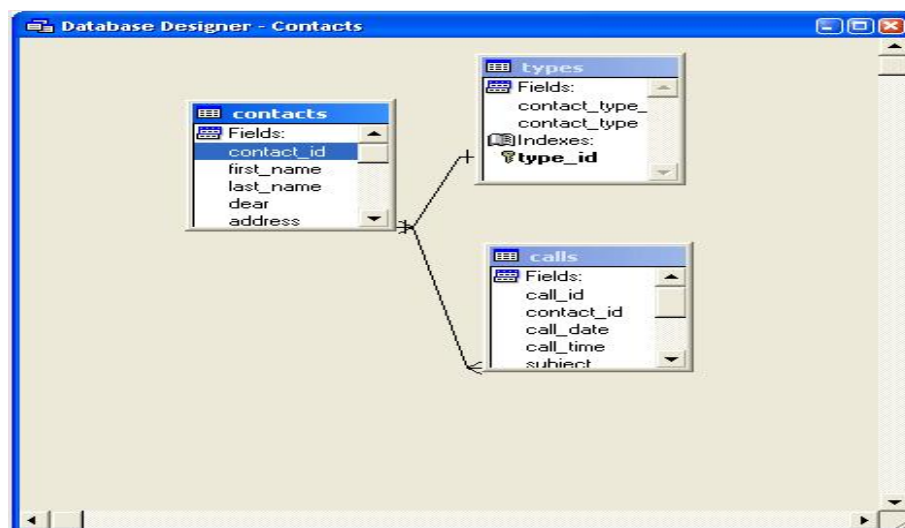


Рисунок 8 - Установление взаимосвязи между таблицами

Тип взаимосвязи между таблицами устанавливается автоматически, в зависимости от типа индексов. Если оба индекса (в родительской таблице и дочерней таблице) уникальные, то возникает тип взаимосвязи «один-к-одному». Если в родительской таблице индекс уникальный, а в дочерней таблице индекс регулярный, то получается тип взаимосвязи «один-ко-многим».

Чтобы редактировать существующую взаимосвязь, надо вызвать на экран диалоговую панель Edit Relationship. Для удаления взаимосвязи ее надо предварительно выделить, а затем нажать клавишу Del.

Одним из главных условий создания базы данных является обеспечение условий ссылочной целостности. Выполнение этих условий гарантирует помещение в базу данных только достоверной информации. Ссылочная целостность обеспечивается механизмом каскадных воздействий: каскадное удаление и каскадное редактирование ключевых полей. Каскадное удаление — это одновременное удаление записи из родительской таблицы с удалением соответствующих записей из всех дочерних таблиц. Удаление записи из дочерней таблицы не влечет за собой удаления соответствующей записи из родительской таблицы. Аналогично, при изменении значения поля (полей) первичного ключа, необходимо изменить значение соответствующего поля (полей) внешнего ключа дочерней таблицы (или таблиц), то есть каскадное редактирование.

Для установления каскадных воздействий надо выделить линию взаимосвязи щелчком левой кнопки мыши и вызвать на экран контекстное меню, щелкнув правой кнопки мыши, когда курсор удерживается на линии связи. В появившемся на экране контекстном меню надо выбрать команду Referential Integrity. На экран выводится диалоговая панель Referential Integrity Builder (рисунок 9).

В верхней части диалоговой панели Referential Integrity имеются графы:

- Parent Table — содержит имена родительских таблиц. Указателем черным треугольником) отмечена текущая родительская таблица;

- Child Table — содержит имена дочерних таблиц, которые могут быть соединены с соответствующими родительскими таблицами;
- Update — содержит один из возможных типов каскадных воздействий при редактировании поля (полей) первичного ключа у соответствующей пары таблиц (имена таблиц указаны в левых графах);
- Delete — содержит один из возможных типов каскадных воз действий при удалении записи из родительской таблицы;
- Insert — содержит один из возможных типов каскадных воз действий при вставке записи в родительскую таблицу;
- Parent Tag — содержит имя тега в родительской таблице. В графе справа (Child Tag) будет указано имя соответствующего тега дочерней таблицы;
- Child Tag — содержит имя тега в дочерней таблице. В графе слева (Parent Tag), будет указано имя соответствующего тега родительской таблицы.

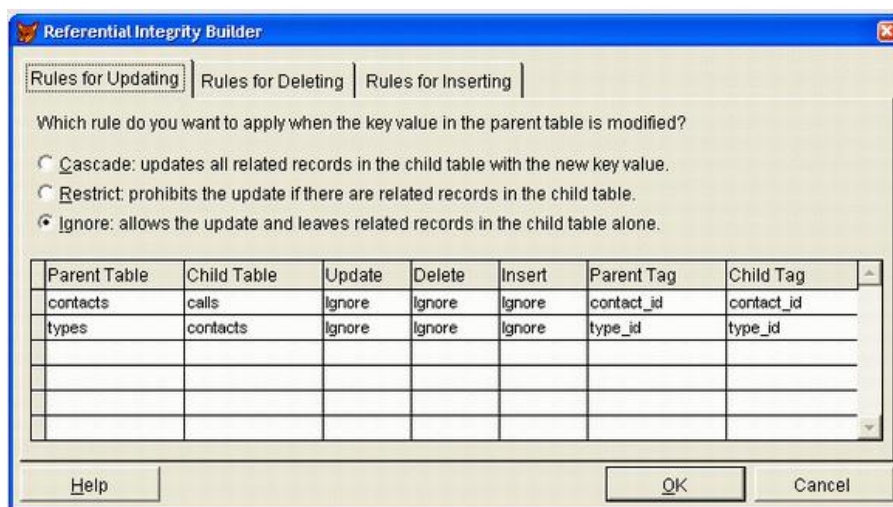


Рисунок 9 - Задание условий ссылочной целостности

Установить соответствующий тип каскадных воздействий можно либо в верхней части диалоговой панели, либо в нижней части диалоговой панели.

В верхней части диалоговой панели в одной из граф Update, Delete или Insert курсором мыши выделяют нужное поле (клетку). В поле появляется кнопка раскрывающегося списка, в котором можно выбрать одно из значений:

Cascade — при изменении значения поля (полей) первичного ключа или альтернативного ключа (Candidate) в родительской таблице автоматически выполняется каскадное изменение значения соответствующего поля (полей) внешнего ключа дочерней таблицы (таблиц).

Restrict — запрещает изменение значения поля (полей) первичного или альтернативного ключа в родительской таблице, если в дочерней таблице (таблицах) имеются соответствующие значения во внешнем ключе.

Ignore — допускаются любые изменения в значениях первичного и альтернативного ключей в родительской таблице. Условия ссылочной целостности не контролируются.

Аналогично работают с графами Delete и Insert, но для графы Insert возможны только два воздействия: Restrict и Ignore. В нижней части диалоговой панели имеются три вкладки:

Rules for Updating — управление каскадным редактированием записей;

Rules for Deleting — управление каскадным удалением записей;

Rules for Inserting — управление каскадным добавлением записей.

Тип воздействия Cascade, Restrict и Ignore устанавливается с помощью соответствующей радиокнопки.

2.4 Практическая часть

2.4.1 Варианты заданий

1) Создать базу данных Авиакомпания, добавить в нее табличные файлы Самолет (номер самолета, модель самолета, технические характеристики самолета), Рейс (номер рейса, пункт отправления, пункт прибытия, время полета, номер самолета), Билет (номер билета, ФИО пассажира, цена билета, номер рейса). При использовании команды Browse задать заголовок «Таблица - Билет». Установить взаимосвязи Самолет-Рейс (1:М) и Рейс-Билет (1:М);

2) Создать базу данных Компьютерные лаборатории, добавить в нее табличные файлы Кабинет (номер кабинета, название кабинета, количество машин в классе), Компьютер (серийный номер компьютера, технические характеристики компьютера), Ответственный (личный номер ответственного за класс, ФИО ответственного, телефон ответственного). При использовании команды Browse таблицы Кабинет разрешить внесение изменений только в поле название кабинета. Установить взаимосвязи Кабинет-Компьютер (1:М) и Ответственный-Кабинет (1:М);

3) Создать базу данных Детский сад, добавить в нее табличные файлы Воспитатель (личный номер воспитателя, ФИО воспитателя, образование воспитателя), Ребенок (ФИО ребенка, дата рождения ребенка, № свидетельства о рождении ребенка), Группа (№ группы, название группы, численность). При использовании команды Browse отобразить левую часть разделенного окна Browse в режиме Edit. Установить взаимосвязи Группа-Воспитатель (1:М) и Группа-Ребенок (1:М);

4) Создать базу данных Бюро путешествий, добавить в нее табличные файлы Туристическое агентство (№ лицензии туристического агентства, телефон, адрес, ФИО директора), Путевка (код путевки, срок действия, дата начала действия, цена, место отдыха, форма отдыха), Клиент (ФИО клиента, № паспорта клиента, телефон клиента). При использовании команды Browse запретить добавление новых записей в конец табличного файла Путевка. Установить взаимосвязи Туристическое агентство - Путевка (1:М) и Путевка -Клиент (1:М);

5) Создать базу данных Искусство, добавить в нее табличные файлы Работа (код работы, название работы, цена работы), Автор (код автора, ФИО

автора, телефон автора), Выставка (название выставки, телефон, адрес, дата выставки). При использовании команды Browse запретить пометку записей к удалению. Установить взаимосвязи Автор - Работа (1:M) и Выставка - Работа (1:M);

6) Создать базу данных Курсы, добавить в нее табличные файлы Курсы (№ лицензии, наименование курсов, продолжительность обучения, дата начала), Преподаватель (код преподавателя, ФИО преподавателя, стаж преподавателя, телефон преподавателя), Слушатель (ФИО слушателя, паспорт слушателя, телефон слушателя). При использовании команды Browse запретить внесение любых изменений в таблице Курсы. Установить взаимосвязи Курсы - Преподаватель (1:M) и Курсы - Слушатель (1:M);

7) Создать базу данных Учебный процесс, добавить в нее табличные файлы Студент (личный номер студента, ФИО студента, дата рождения, группа), Дисциплина (код дисциплины, название дисциплины, количество часов), Преподаватель (код преподавателя, ФИО преподавателя). При использовании команды Browse установить для правой части разделенного окна Browse режим просмотра Edit. Установить взаимосвязи Дисциплина - Преподаватель (1:M) и Студент - Дисциплина (M:M);

8) Создать базу данных Товар, добавить в нее табличные файлы Производитель (код производителя, название производителя, адрес производителя), Товар (код товара, характеристика товара, дата выпуска, срок годности), Магазин (№ магазина, название магазина, адрес магазина, телефон магазина). При использовании команды Browse задать заголовок «Таблица - Магазин». Установить взаимосвязи Товар - Производитель (1:M) и Товар - Магазин (M:M);

9) Создать базу данных Производство, добавить в нее табличные файлы Деталь (код детали, наименование детали, размер, характеристика), Исполнитель (№ исполнителя, ФИО исполнителя, адрес исполнителя), Договор № договора, дата подписания, количество деталей, срок исполнения). При использовании команды Browse таблицы Исполнитель разрешить внесение изменений только в поле адрес исполнителя. Установить взаимосвязи Деталь - Договор (1:M) и Исполнитель - Договор (1:M);

10) Создать базу данных Автомобили, добавить в нее табличные файлы Автомобиль (№ автомобиля, марка автомобиля, дата выпуска, технические характеристики, цвет), Владелец (паспорт владельца, ФИО владельца, адрес владельца), Технический осмотр автомобиля (№ тех. осмотра, дата, срок). При использовании команды Browse отобразить левую часть разделенного окна Browse в режиме Edit. Установить взаимосвязи Владелец - Автомобиль (1:M) и Автомобиль - Технический осмотр автомобиля (M:M);

11) Создать базу данных Спортивный клуб, добавить в нее табличные файлы Клуб (№ лицензии клуба, название клуба, адрес, телефон), Тренер (код тренера, ФИО тренера, телефон тренера), Посетитель (код посетителя, ФИО посетителя, телефон посетителя). При использовании команды Browse запретить добавление новых записей в конец табличного файла Клуб. Установить взаимосвязи Клуб - Тренер (1:M) и Клуб - Посетитель (M:M);

12) Создать базу данных Соревнования, добавить в нее табличные файлы Команда (№ команды, название команды, количество игроков), Тренер (код тренера, ФИО тренера, награды), Соревнования (кол соревнований, дата соревнований, исход соревнований). При использовании команды Browse запретить пометку записей к удалению. Установить взаимосвязи Команда - Тренер (1:1) и Соревнования - Команда (М:М).

2.4.2 Порядок выполнения работы

1) В окне Command установить германский формат даты. На экран вывести статус строку;

2) Согласно своему варианту, создать три таблицы базы данных двумя способами: с помощью команды USE и с помощью главного меню;

3) Заполнить каждую из таблиц не менее чем пятью записями;

4) Отобразить на экране первую таблицу в режиме BROWSE. При этом с помощью опции FREEZE команды BROWSE разрешить внесение изменений только в одно указанное поле;

5) Внести небольшие изменения (например, изменить тип поля или размер поля) в структуру таблицы;

6) Вывести содержимое первой таблицы в файл с расширением .txt в свою папку с помощью команды LIST;

7) Отобразить на экране вторую таблицу в режиме CHANGE и запретить внесение любых изменений в таблице с помощью опции NOEDIT;

8) Вывести содержимое второй таблицы в файл с расширением .txt в свою папку с помощью команды DISPLAY;

9) Третью таблицу представить в виде разделенного окна: в правой его части данные выводятся в режиме CHANGE, а в левой в режиме BROWSE;

10) Добавить три записи в таблицу с помощью команды APPEND;

11) Затем пометить их к удалению с помощью клавиатуры, с помощью мыши и с помощью команды DELETE. Физически удалить помеченные записи;

12) С помощью главного меню и Command-окна добавить все три таблицы в одну базу данных;

13) Установить взаимосвязи между таблицами;

14) Обеспечить условия ссылочной целостности базы данных через механизм каскадных воздействий;

15) Сделать вывод о проделанной работе. Оформить отчет.

2.4.3 Содержание отчета

1) тема, цель;

2) постановка задачи, вариант задания;

3) описание выполнения работы;

4) содержимое Command-окна;

- 5) содержимое табличных файлов (в режиме Browse и Edit);
- 6) диалоговая панель Database Designer (с установленными взаимосвязями);
- 7) вывод.

2.4.4 Вопросы для защиты работы

- 1) Какие есть способы создания таблиц БД?
- 2) Какая опция команды USE делает открываемую таблицу, предназначенной для личного использования при работе в локальной сети?
- 3) Какая команда предоставляет широкие возможности для доступа к данным: просмотр, редактирование, добавление и удаление?
- 4) Какая опция команды BROWSE разрешает внесение изменений только в одно указанное поле, хотя на экран выводятся все поля?
- 5) Какая опция команды BROWSE запрещает внесение любых изменений в таблице?
- 6) С помощью какой команды можно выполнить физическое удаление помеченных записей?
- 7) Какие существуют способы добавления таблиц в базу данных?
- 8) С помощью каких команд можно вывести записи на экран (в файл, на принтер)? В чем отличие этих команд?
- 9) Как установить взаимосвязь между таблицами 1:1?
- 10) Как установить взаимосвязь между таблицами 1:M?

3 Управление базами данных

3.1 Цель работы

Научится применять способы сортировки, фильтрации и поиска данных в базах данных.

3.2 Постановка задачи

В созданной базе данных выполнить сортировку, поиск и фильтрацию данных, согласно своему варианту.

3.3 Теоретическая часть

3.3.1 Сортировка данных

Сортировка данных — это их упорядочение по значениям одного или нескольких полей. Упорядочение может быть выполнено как по возрастанию, так и по убыванию.

В FoxPro сортировка производится следующими способами:

1) В соответствии с индексом.

Сначала индекс делают текущим, а затем выводят таблицу на экран или обновляют содержимое экрана. Если построен сложный индекс, то и сортировка будет выполнена по полям, входящим в индекс. Это наиболее удобный вид сортировки, так как для своего выполнения требует мало времени, не требует дополнительных затрат памяти и исключена потеря информации.

2) С помощью команды SORT

В этом случае сортировку таблицы можно произвести по любому полю (или полям), даже если по указанному полю (или полям) нет индекса.

```
SORT TO <имя файла> ON <имя поля 1> [/A] [/D] [/C]
```

```
[,<имя поля 2> [/A] [/D] [/C]... ]
```

```
[ASCENDING | DESCENDING]
```

```
[SCOPE] [FOR <выр.L1>] [WHILE <выр.L2>]
```

```
[FIELDS <список имен полей>][NOOPTIMIZE]
```

Назначение опций:

<имя файла> — имя табличного файла (таблицы), который надо отсортировать.

ON ... — указывается одно или несколько полей, по значениям которых надо выполнить сортировку таблицы. Каждое поле может иметь уточняющие параметры:

/A — сортировка по возрастанию. Этот параметр принят по умолчанию. /D — сортировка по убыванию;

/C — игнорирует значение регистра для символьных полей (строчные и прописные буквы считаются одинаковыми);

ASCENDING — задает сортировку по возрастанию только для полей, не имеющих параметров /A и /D;

DESCENDING — задает сортировку по убыванию только для полей, не имеющих параметров /A и /D;

FIELDS — в списке перечисляются имена полей, которые надо поместить во вновь создаваемый табличный файл. Если опция опущена, то в создаваемый файл помещаются все поля из исходного табличного файла.

При использовании этой команды на диске дополнительно сохраняется отсортированный файл, на построение которого требуется дополнительное время. Эта команда используется редко.

3.3.2 Поиск данных

Поиск данных часто используемая функция при работе с базами данных.

В FoxPro предусмотрены две методики поиска данных: метод последовательного перебора и метод деления пополам.

При поиске методом полного перебора просматривается вся таблица от первой записи до последней.

Поиск производится по любому полю таблицы.

```
LOCATE FOR [SCOPE] <условие поиска>  
[WHILE <выр.L> ] [NOOPTIMIZE]
```

В <условие поиска> — пишется имя поля для поиска значения, далее указывается один из логических знаков <>, <=, >=, = или <> и само искомое значение. Если искомое значение имеет символьный тип, то оно указывается в двойных кавычках. Если искомое значение имеет тип дата, то оно указывается в апострофах. Если искомое значение имеет числовой тип, то выделять его знаками не требуется.

Если поиск закончился успешно, то курсор устанавливается на найденную запись. Для того чтобы увидеть найденную запись, надо подать команду DISPLAY или BROWSE. Для нахождения следующей записи, по тому же искомому значению, надо подать команду CONTINUE. Если поиск закончился неудачно, то курсор устанавливается на последнюю запись и на экран выводится сообщение: End of Locate Scope.

Этот поиск можно выполнить с помощью команды Главного меню: Table/Go to Record /Locate.

На экран выводится диалоговая панель Locate Record (рисунок 10).

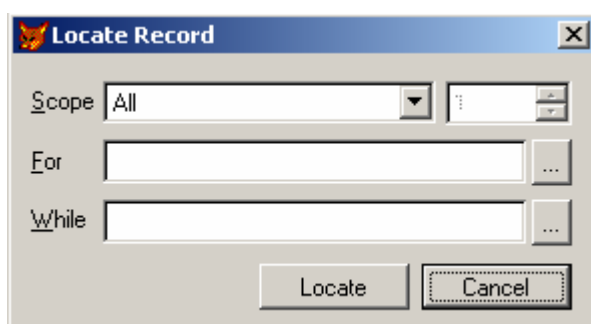


Рисунок 10 - Диалоговая панель Locate Record

Назначение элементов окна:

For — выводит на экран диалоговую панель для задания условия поиска;

Scope и While — выводят на экран диалоговые панели для задания ограничений количества просматриваемых записей.

В случае успешного и неуспешного поиска курсор принимает соответствующее положение. Для нахождения следующей записи, соответствующей условию поиска, надо повторить эту процедуру еще раз. Если

по полю поиска построен индекс, то автоматически будет использоваться команда SEEK.

Поиск по полю текущего индекса выполняется следующими способами:

1) Поиск с помощью окна Command. В этом случае надо воспользоваться одной из команд:

SEEK <искомое значение> или *FIND* <искомое значение>.

2) Поиск с помощью команды Главного меню

Предварительно таблицу, в которой надо найти какое-либо значение, делают текущей. Затем из Главного меню подают команду Edit / Find. На экран выводится диалоговая панель Find (рисунок 11).

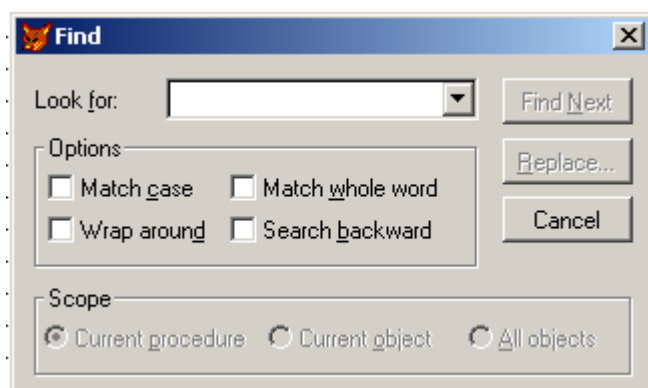


Рисунок 11 - диалоговая панель Find

В поле ввода Look For вводится значение для поиска. При обнаружении первого значения запись делается текущей, то есть на нее устанавливается указатель (черный треугольник слева от первого поля).

Флажки в группе Options задают следующие условия поиска:

- Match Case — различие регистров. Прописные и строчные буквы считаются разными;

- Wrap Around — круговой поиск. При достижении конца таблицы поиск начинается заново с первой строки;

- Match Whole Word — точное соответствие полному значению поиска.

При выключенном флажке значение поиска может входить как часть в найденное слово;

- Search Backward — поиск в обратном направлении (снизу вверх).

Кнопка Find Next предназначена для нахождения следующего значения, удовлетворяющего условию поиска.

3.3.3 Фильтрация данных

В FoxPro определены два вида фильтров данных: фильтр для строк, когда ограничивается количество строк, предъявляемых на экран, и фильтр для полей, когда ограничивается количество полей, отображаемых на экране и, следовательно, доступных для редактирования.

Для установки фильтра данных используют команду:

```
SET FILTER TO <выр. L>
```

В опции <выр. L> указывают имя поля и его значение, по которым надо выполнить фильтрацию. Для снятия фильтра и восстановления первоначального вида таблицы используют ту же команду, но без опции. Для определения списка полей используют команду:

```
SET FIELDS TO [<список полей> | ALL [LIKE <маска> | EXCEPT <маска>] ]
```

После выполнения этой команды, из текущей таблицы для установки фильтра будут отобраны либо поля, имена которых указаны в <списке полей>, либо все поля. По умолчанию принята опция ALL — все поля. Опция LIKE включает в список полей все поля, имена которых удовлетворяют маске (шаблону). Опция EXCEPT включает в список полей все поля, имена которых не удовлетворяют маске (шаблону).

Если повторно подать эту же команду с другим <списком полей>, то вновь определенный список полей будет добавлен к существующему списку полей. Команда *SET FIELDS TO* без опции закрывает все поля, открытые предыдущими командами *SET FIELDS TO*.

Для установки фильтра полей используют команду

```
SET FIELDS ON | OFF | LOCAL | GLOBAL.
```

Назначение опций:

ON — устанавливает фильтр для ранее определенных полей;

OFF — отменяет список полей и разрешает отображение всех полей таблицы. По умолчанию установлена опция OFF;

LOCAL — определяет, что для фильтра доступны только поля текущей таблицы;

GLOBAL — разрешает отображать поля всех таблиц, между которыми установлены реляционные отношения (установлены взаимосвязи).

3.4 Практическая часть

3.4.1 Варианты заданий

1) В таблице «Самолет» выполнить сортировку записей по полю номер самолета с помощью команды Sort. В таблице «Рейс» выполнить сортировку записей по полю № рейса в соответствии с индексом. В таблице «Билет» выполнить поиск записи, в которой № билета 100. Вывести записи

только тех пассажиров, у которых фамилия начинается на букву А. В таблице «Рейс» выполнить фильтрацию по полям : № рейса и время полета;

2) В таблице «Кабинет» выполнить сортировку записей по полю название кабинета с помощью команды Sort. В таблице «Компьютер» выполнить сортировку записей по полю серийный номер компьютера в соответствии с индексом. В таблице «Компьютер» выполнить поиск записи, в которой серийный номер компьютера 110011. Вывести записи только тех ответственных, у которых имя начинается на букву Б. В таблице «Кабинет» выполнить фильтрацию по полям : № кабинета и количество ПК;

3) В таблице «Воспитатель» выполнить сортировку записей по полю фамилия воспитателя с помощью команды Sort. В таблице «Группа» выполнить сортировку записей по полю № группы в соответствии с индексом. В таблице «Ребенок» выполнить поиск записи, в которой № домашнего телефона 776655. Вывести записи только тех Воспитателей, у которых высшее образование. В таблице «Группа» выполнить фильтрацию по полям : № группы и количество детей в группе;

4) В таблице «Туристическое агентство » выполнить сортировку записей по полю номер лицензии агентства с помощью команды Sort. В таблице «Путевка» выполнить сортировку записей по полю № путевки в соответствии с индексом. В таблице «Клиент» выполнить поиск записи, в которой № паспорта 11223344. Вывести записи только тех клиентов, у которых фамилия начинается на букву А. В таблице «Путевка» выполнить фильтрацию по полям : № путевки и место отдыха;

5) В таблице «Работа» выполнить сортировку записей по полю название работы с помощью команды Sort. В таблице «Музей» выполнить сортировку записей по полю номер лицензии музея в соответствии с индексом. В таблице «Автор» выполнить поиск записи, в которой № телефона автора 665544. Вывести записи только тех работ, цена которых начинается от 100000 рублей. В таблице «Музей» выполнить фильтрацию по полям : название музея и телефон музея;

6) В таблице «Курсы» выполнить сортировку записей по полю наименование курсов с помощью команды Sort. В таблице «Преподаватель» выполнить сортировку записей по полю фамилия преподавателя в соответствии с индексом. В таблице «Слушатель» выполнить поиск записи, в которой фамилия слушателя Сидоров. Вывести записи только курсов, продолжительность которых 100 часов. В таблице «Слушатель» выполнить фильтрацию по полям : фамилия слушателя и номер телефона слушателя;

7) В таблице «Студент» выполнить сортировку записей по полю фамилия студента с помощью команды Sort. В таблице «Дисциплина» выполнить сортировку записей по коду дисциплины в соответствии с индексом. В таблице «Преподаватель» выполнить поиск записи, в которой фамилия преподавателя Иванов. Вывести записи только тех дисциплин, количество часов которых более 100 часов. В таблице «Дисциплина» выполнить фильтрацию по полям : название дисциплины и количество часов;

8) В таблице «Производитель» выполнить сортировку записей по полю код производителя с помощью команды Sort. В таблице «Товар» выполнить

сортировку записей по полю код товара в соответствии с индексом. В таблице «Магазин» выполнить поиск записи, в которой телефон магазина 110022. Вывести записи только товаров, наименование которых начинаются на букву П. В таблице «Магазин» выполнить фильтрацию по полям: название магазина и телефон магазина;

9) В таблице «Деталь» выполнить сортировку записей по полю наименование детали с помощью команды Sort. В таблице «Договор» выполнить сортировку записей по полю № договора в соответствии с индексом. В таблице «Исполнитель» выполнить поиск записи, в которой фамилия исполнителя - Иванов. Вывести записи только деталей, наименование которых втулка. В таблице «Договор» выполнить фильтрацию по полям: № договора и срок исполнения;

10) В таблице «Автомобиль» выполнить сортировку записей по полю марка автомобиля с помощью команды Sort. В таблице «Владелец» выполнить сортировку записей по полю номер паспорта владельца в соответствии с индексом. В таблице «Тех. паспорт» выполнить поиск записи, в которой номер тех. паспорта 110022. Вывести записи только владельцев, фамилии которых начинаются на букву П. В таблице «Тех. паспорт» выполнить фильтрацию по полям: номер тех. паспорта и срок действия;

11) В таблице «Спортивный клуб» выполнить сортировку записей по полю название с помощью команды Sort. В таблице «Тренер» выполнить сортировку записей по полю номер паспорта тренера в соответствии с индексом. В таблице «Посетитель» выполнить поиск записи, в которой номер телефона посетителя 770022. Вывести записи только тренеров, разряд которых 14. В таблице «Посетитель» выполнить фильтрацию по полям: фамилия и номер телефона;

12) В таблице «Соревнования» выполнить сортировку записей по полю номер с помощью команды Sort. В таблице «Команда» выполнить сортировку записей по полю название команды в соответствии с индексом. В таблице «Тренер» выполнить поиск записи, в которой номер телефона тренера 775522. Вывести записи только команд, количество игроков в которых 10. В таблице «Команда» выполнить фильтрацию по полям: название и рейтинг.

3.4.2 Порядок выполнения работы

- 1) согласно заданию по своему варианту добавить необходимые записи в таблицы;
- 2) выполнить сортировку с помощью команды *SORT*;
- 3) выполнить сортировку по полю текущего индекса, предварительно создав необходимые индексы;
- 4) выполнить поиск записей с помощью команд *LOCATE* и *SEEK (FIND)*;
- 5) выполнить фильтрацию по строкам и столбцам;
- 6) сделать вывод о проделанной работе;
- 7) оформить отчет.

3.4.3 Содержание отчета

- 1) тема, цель;
- 2) постановка задачи, вариант задания;
- 3) порядок выполнения работы;
- 4) результаты работы;
- 5) вывод.

3.4.4 Вопросы для защиты работы

- 1) Укажите два способа сортировки данных в таблице.
- 2) Укажите особенности применения команды SORT.
- 3) Для какой части табличного файла можно применить сортировку? Как задать часть табличного файла, для которой надо выполнить сортировку?
- 4) Укажите два способа поиска данных в таблице.
- 5) Какими особенностями обладает команда LOCATE?
- 6) Какими особенностями обладает команда SEEK?
- 7) Как задать ограничения на количество строк при фильтрации данных?
- 8) Как задать ограничения на количество полей табличного файла при фильтрации данных?

4 Работа с программным файлом

4.1 Цель работы

Научится создавать программные файлы в Visual Fox Pro. Научиться применять команды и конструкции для программирования.

4.2 Постановка задачи

Создать программный файл. Реализовать в нем функциональность согласно своему варианту.

4.3 Теоретическая часть

4.3.1 Создание программного файла

Программный файл создается с помощью встроенного текстового редактора FoxPro. Текстовый редактор можно запустить как из окна Command, так и из Главного меню.

Чтобы запустить текстовый редактор из окна Command, в этом окне надо задать команду:

```
MODIFY COMMAND | FILE [<имя файла> | ?]  
[NOEDIT]  
[NOMENU]  
[NOWAIT]  
[RANGE , ]  
[[WINDOW <имя окна 1>] [IN [WINDOW] <имя окна 2> | IN SCREEN ]  
[SAME]  
[SAVE]
```

Ключевое слово COMMAND предназначено как для создания нового программного файла (имя файла не указывается) с расширением .prg, так и для открытия уже существующего файла (имя файла указывается либо полностью, либо частично — по шаблону).

Ключевое слово FILE предназначено только для открытия существующего программного файла, причем имя файла надо указывать полностью.

Если имя файла не известно, то вместо имени файла можно указать символ «?» и на экран будут выведен список имен программных файлов (с расширением .prg), где можно выбрать имя нужного файла.

Назначение опций:

NOEDIT — разрешает просмотр текста, но редактирование текста запрещено;

NOMENU — запрещает вызов системного меню окна редактирования, то есть блокирует клавишу F10 и кнопку в левом верхнем углу окна редактирования;

NOWAIT — после открытия окна редактирования разрешает работу (без пауз) программы, вызвавшей открытие окна редактирования. В противном случае работа программы прерывается до закрытия окна редактирования.

RANGE , <выр.N2> — при работе с Метод-полем позволяет выбрать часть текста в указанном интервале от до;

WINDOW <имя окна 1> — открывает окно редактирования с характеристиками другого предварительно открытого окна <имя окна 1>;

IN WINDOW <имя окна 2> — открывает окно редактирования внутри окна <имя окна 2>;

IN SCREEN — опция задана по умолчанию. Открывает окно редактирования внутри главного окна FoxPro.

SAME — если текст программы, текст документа или Мемо-поле уже открыты, то запрещает новое редактирование в окне редактора.

SAVE — опция используется только в программах и сохраняет на экране образ окна редактирования после его закрытия.

Для создания нового программного файла из Главного меню надо подать команду: File/New и выбрать расширение .prg (или Program).

4.3.2 Открытие и запуск программного файла

Для открытия уже существующего программного файла надо подать команду: File/Open и указать имя нужного программного файла. Редактирование текста программного файла производится по правилам текстовых редакторов.

Выполнить программный файл можно либо из окна Command с помощью команды DO <имя программного файла>, либо с помощью команды Главного меню: PROGRAM / DO, далее указать в списке имя программного файла, либо нажав в графическом меню кнопку, на которой изображен восклицательный знак.

Остановить выполнение программного файла можно нажатием клавиши Esc, если в тексте программного файла указана команда SET ESCAPE ON. Во многих версиях FoxPro эта команда указана по умолчанию.

4.4 Практическая часть

4.4.1 Варианты заданий

- 1) Вычислить сумму в рублях, вырученную со всех билетов одного рейса;
- 2) Рассчитать для каждого класса процент укомплектованности компьютерами (10 машин – норма);
- 3) Рассчитать процент воспитателей с высшим образованием в детском саду;
- 4) Вычислить сумму (в руб.), вырученную со всех путевок одного места отдыха;
- 5) Подсчитать процент картин одного автора от общего количества картин в музее;
- 6) Рассчитать дату окончания всех курсов;
- 7) Рассчитать процент специальных дисциплин, от общего количества изучаемых дисциплин;
- 8) Вывести наименования товаров с истекшим сроком годности.
- 9) Вычислить процент деталей одного наименования, выполненного одним исполнителем, от общего количества деталей;
- 10) Вывести дату прохождения следующего тех.осмотра для каждого автомобиля;

11) Подсчитать количество тренеров, получивших награды за последние два года;

12) Вывести количество побед для каждой команды.

4.4.2 Порядок выполнения работы

1) Создать структуру базы данных «Заработная плата». БД должна содержать следующие сведения: ФИО, Оклад, Уральский коэффициент, Премия, Аванс, Всего начислено, Подоходный налог, Отчисления в пенсионный фонд, Всего удержано, Всего к оплате;

2) Заполнить поля ФИО, Оклад (5 записей); поля Премия и Аванс в нескольких произвольных записях; все остальные поля оставить нулевыми;

3) Создать программный файл (Modify command) для расчета заработной платы. Расчет необходимо производить следующим образом:

– Уральский коэффициент составляет 15% от оклада;

– Всего начислено = Оклад + Ур.Коэф. + Премия;

– Подоход.налог = 12% от (Всего начисл. – Миним.з.п.);

– Пенс.фонд = 2% от всей начисленной суммы;

– Всего к оплате = Всего начисл. – Всего удержано – Аванс;

4) Выполнить программу (Do <имя программного файла>);

5) Создать программный файл, соответственно своему варианту;

6) Сделать вывод о проделанной работе. Оформить отчет.

4.4.3 Содержание отчета

1) тема, цель;

2) постановка общей задачи (для расчета заработной платы);

3) текст программного файла для общего задания;

4) результаты работы программного файла;

5) вариант задания;

6) текст программного файла для задания по варианту;

7) результаты работы программного файла;

8) содержимое Command-окна;

9) вывод.

4.4.4 Вопросы для защиты работы

1) Укажите назначение программного файла.

2) Укажите несколько способов создания программного файла.

3) Как открыть программный файл?

- 4) Как поставить на выполнение программный файл?
- 5) Назовите команды вывода.
- 6) Как задать комментарий в тексте программы?
- 7) По каким правилам задаются имена переменных?
- 8) Как определяется тип переменной?
- 9) Как получить сведения о структуре таблицы и вывести их на экран?
- 10) Укажите назначение и особенности команд ? и ??.

5 Создание меню с помощью конструктора меню

5.1 Цель работы

Научиться создавать меню различных типов с помощью конструктора меню.

5.2 Постановка задачи

Создать горизонтальное меню и всплывающее меню с помощью конструктора меню Visual Fox Pro.

5.3 Теоретическая часть

5.3.1 Действия, необходимые для создания меню

Строкой меню называется горизонтальное меню, располагаемое в верхней части экрана. Примером строки меню является основное меню Visual FoxPro, а также меню программ, работающих в среде Windows. Созданное вами в конструкторе меню может замещать основное меню Visual FoxPro или добавляться к нему.

Для создания меню необходимо выполнить следующие действия:

- 1) Открыть окно конструктора меню;
- 2) Описать вид меню, текст, пункты меню и его атрибуты;
- 3) Определить действия, которые будут выполняться при выборе пунктов меню;
- 4) Сгенерировать меню, используя команду Generate (Генерация) из меню Menu (Меню). При этом создается программа, которую вы в результате и запускаете на выполнение.

Диалоговое окно New Menu (Новое меню) предлагает два варианта меню:
- Menu — меню в виде строки;

- Shortcut — всплывающее меню, в котором основные пункты расположены по вертикали.

5.3.2 Создание меню в виде строки

Сначала рассмотрим создание меню в виде строки. Для этого выберите опцию Menu (Меню).

Область конструктора меню, над которой размещены надписи Prompt (Приглашение), Result (Результат) и Options (Опции), предназначена для формирования меню.

В поле Prompt (Приглашение) можно ввести наименования пунктов меню. Раскрывающийся список Result (Результат) используется для указания типа пункта меню. Кнопка Options (Опции) открывает диалоговое окно Prompt Options (Опции элемента меню), в котором можно определить дополнительные параметры данного элемента меню ("горячие" клавиши, сообщение, отображаемое в строке состояния при выборе пункта меню, и т. д.). В списке Menu level (Уровень меню) указывается уровень текущего меню. Слева в конструкторе меню размещены кнопки (таблица. 3).

Таблица 3 - Назначение кнопок конструктора меню

Кнопка	Назначение
Insert (Вставить)	Добавляет в меню новый пункт
Insert Bar (Вставить команды системного меню)	Открывает диалоговое окно Insert System Menu Bar, содержащее команды системного меню Visual FoxPro, позволяя разместить их в создаваемом пользовательском меню
Delete (Удалить)	Удаляет текущий пункт меню
Preview (Просмотр)	Размещает создаваемое меню на экране, позволяя посмотреть его внешний вид

После того как вы открыли конструктор, можно приступать к созданию меню. Для этого выполните следующие действия:

1) В поле Prompt (Приглашение) введите наименования первого пункта меню и нажмите клавишу <Enter> или <Tab> для перехода на следующее поле. Курсор оказывается в списке Result (Результат).

2) Для определения типа пункта меню (таблица.4) нажмите кнопку раскрытия списка и выберите необходимое значение из тех, которые предлагает система.

Таблица 4 Типы пунктов меню

Тип меню	Назначение
1	2
Command (Команда)	При выборе пункта меню данного типа будет выполняться связанная с ним команда
Pad Name (Наименование строки меню)	При выборе пункта меню никаких действий выполняться не будет. Как правило, используется в качестве дополнительного пояснения к меню
Submenu (Подменю)	При выборе пункта меню раскрывается связанное с данным пунктом ниспадающее меню
Procedure (Процедура)	При выборе пункта меню вызывается процедура, определенная для данного пункта меню

При использовании значения типа Command (Команда) с правой стороны появляется поле для ввода команды, выполняемой при выборе данного пункта меню. Это может быть, например, команда вызова формы или формирования отчета. Если из списка Result (Результат) вы выбрали значение Procedure (Процедура) или Submenu (Подменю), в окне конструктора правее описания типа пункта меню появляется кнопка Create (Создать). При нажатии на эту кнопку вы переходите, соответственно, в окно создания процедуры или в окно создания ниспадающего меню для выбранного пункта меню (рисунок 12).

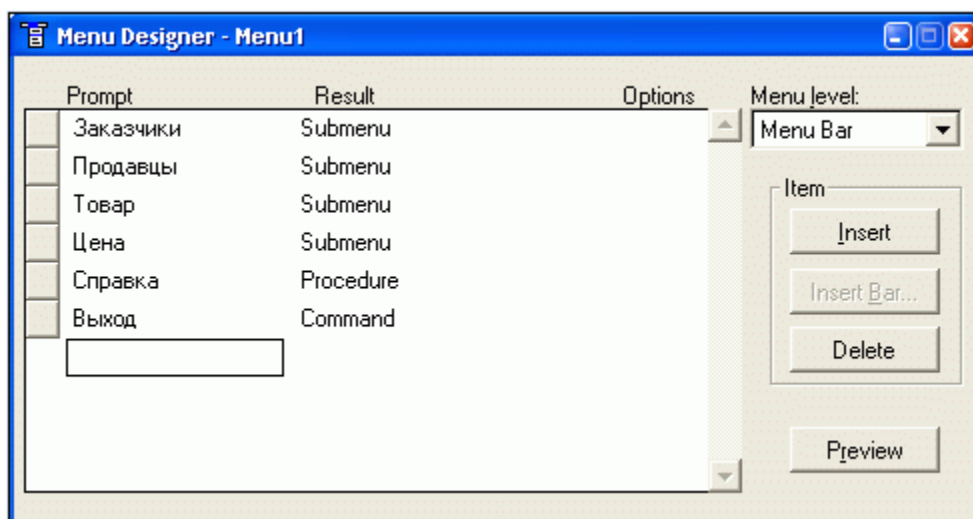


Рисунок 12 - Основное меню приложения

- 3) Указав тип пункта меню, перейдите в следующую строку и введите информацию о втором пункте меню;
- 4) Введите наименования остальных пунктов меню и их типы .
- 5) Для просмотра созданных пунктов меню нажмите кнопку Preview (Просмотр). Основное меню Visual FoxPro будет заменено созданным меню.

Пункты меню отображаются на экране в порядке их описания. На экране также появляется диалоговое окно Preview (Просмотр), в котором отображается текст текущего пункта меню, его тип и выполняемое действие (рисунок 13).



Рисунок 13 - Просмотр созданного меню приложения

Для создания "горячей" клавиши, предоставляющей пользователю возможность ускоренного выбора пункта меню, необходимо включить в его имя перед активизирующим символом следующие символы \<. В качестве "горячей" клавиши можно использовать первый символ имени пункта, что, конечно, более предпочтительно, а также и любой другой символ из его имени. Символы, используемые в качестве "горячей" клавиши, выделяются в строке меню подчеркиванием.

"Горячие" клавиши для пунктов меню назначаются Visual FoxPro по умолчанию. Для их создания используются первые буквы элементов строки меню. Если два элемента вашего меню начинаются с одинаковой буквы, то обоим элементам строки меню в качестве "горячей" клавиши назначается одинаковый символ. В этом случае вам нужно переопределить "горячую" клавишу для одного из элементов строки меню (рисунок 14).

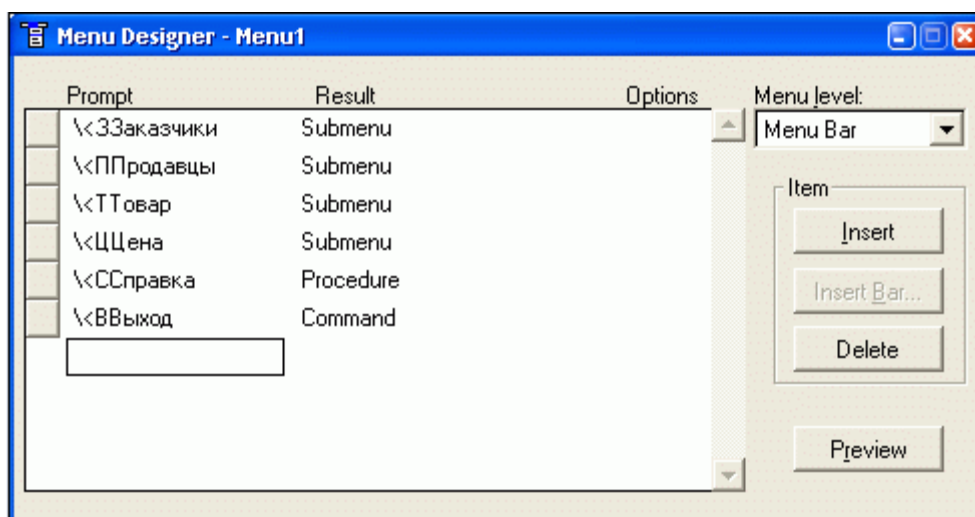


Рисунок 14 - Создание строки меню с назначенными "горячими" клавишами

Для задания параметров пункта меню используется диалоговое окно Prompt Options (Опции элемента меню). Для открытия этого окна установите в

конструкторе курсор на пункт меню и нажмите появившуюся в строке кнопку Options (Опции). В диалоговом окне Prompt Options (Опции элемента меню) содержится область Shortcut (Всплывающее меню) для задания клавиш быстрого вызова, а также поля ввода (таблица 5).

Таблица 5 - Назначение полей ввода диалогового окна Prompt Options

Поле ввода	Назначение
Skip For (Пропустить для)	Позволяет заблокировать пункт меню
Message (Сообщение)	Позволяет задать сообщение, которое будет отображаться в строке состояния при выборе пункта меню
Pad Name (Имя пункта меню)	Задаёт имя пункта меню
Comment (Комментарий)	Задаёт комментарий к пункту меню

Область Negotiate (Соглашение) содержит два раскрывающихся списка, имеющих следующее назначение:

- Container (Контейнер) — определяет расположение меню при редактировании по месту OLE-объектов;
- Object (Объект) — задает расположение меню при выполнении приложения типа Active Document в Web-браузере.

Рассмотрим последовательность действий по определению клавиши быстрого вызова пункта меню справочной системы приложения:

- 1) в конструкторе меню установите курсор на пункт меню Справка;
- 2) нажмите кнопку Options (Опции);
- 3) в открывшемся диалоговом окне Prompt Options (Опции элемента меню) в области Shortcut (Всплывающее меню) поле Key Label (Метка) содержит подсказку press the key (нажмите клавишу). Установите курсор в поле Key Label (Метка) и нажмите клавишу или комбинацию клавиш, используемую в качестве клавиши быстрого вызова, и в поле появится метка, соответствующая нажатию клавиши. В поле ввода Key Text (Пояснение) вы можете ввести краткое пояснение к определяемой клавише. Для пункта меню Справка нажмите клавишу <F1> клавиатуры. В поле Key Text (Пояснение) по умолчанию будет введено F1 (рисунок 15). Оставьте это значение.

- 4) Нажмите кнопку ОК для закрытия диалогового окна.

Используя поле ввода Message (Сообщение) диалогового окна Prompt Options (Опции элемента меню), можно определить для пункта меню сообщение, которое будет отображаться в строке состояния при установке на него курсора. В качестве сообщения можно использовать не только строку текста, но и произвольное символьное выражение. В этом случае нажмите кнопку вызова конструктора выражения, расположенную с правой стороны поля, и в открывшемся диалоговом окне Expression Builder (Построитель выражения) создайте необходимое выражение.

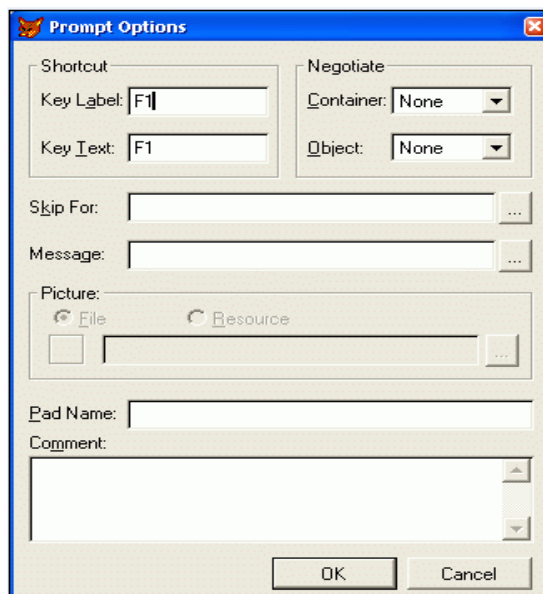


Рисунок 15 - Определение клавиши быстрого вызова для пункта меню Справка

Определим сообщение для пункта меню Справка. Для этого выполните следующие действия:

- 1) нажмите кнопку Options (Опции) для пункта меню Справка;
- 2) в открывшемся диалоговом окне Prompt Options (Опции элемента меню) нажмите кнопку вызова построителя выражения для поля Message (Сообщение);
- 3) в поле Message (Сообщение) построителя выражений введите строку текста "Вызов справочной системы приложения " (введенную строку текста не забудьте поместить в кавычки);
- 4) нажмите кнопку ОК для закрытия диалогового окна Expression Builder (Построитель выражения), Вы оказываетесь в окне Prompt Options (Опции элемента меню), введенное вами выражение размещено в поле Message (Сообщение) (рисунок 16);
- 5) нажмите кнопку ОК.

По умолчанию при генерации программы Visual FoxPro в качестве имен пунктов меню создает уникальные имена. Для повышения читабельности программы можно определить эти имена явным образом, используя поле ввода Pad Name (Имя пункта меню) диалогового окна Prompt Options (Опции элемента меню).

На рисунке 17 приведено диалоговое окно Prompt Options (Опции элемента меню) для пункта меню Справка с определенным для него именем пункта меню, сообщением в строке состояния и условием блокировки.

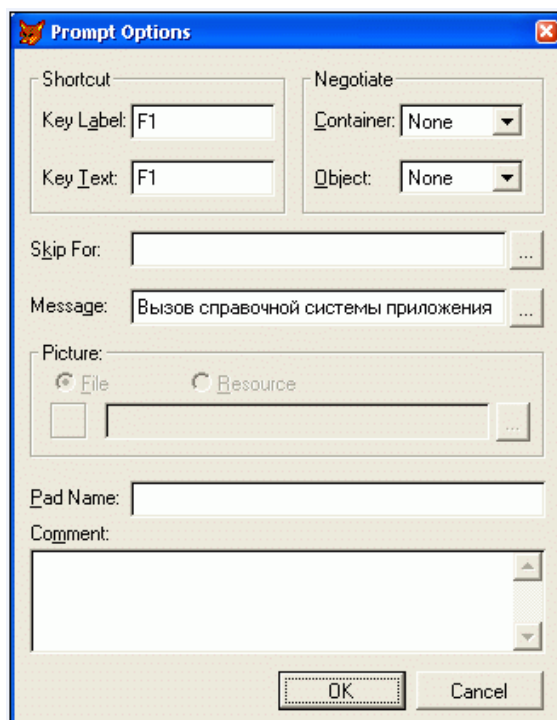


Рисунок 16- Определение сообщения для пункта меню Справка

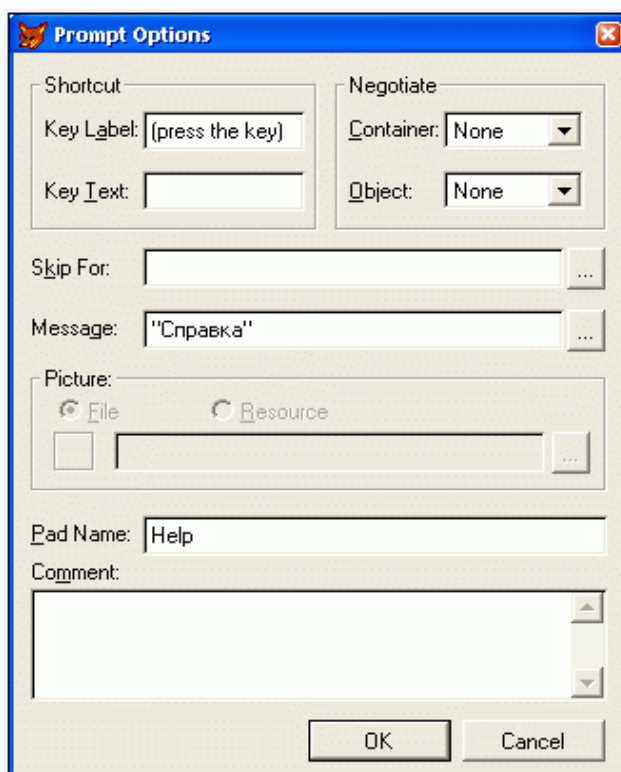


Рисунок 17- Диалоговое окно Prompt Options для пункта меню Справка

Основное назначение пункта меню — выполнять определенное для него действие. Результат выбора пункта меню задается его типом (таблица 6).

Таблица 6 - Типы пункта меню

Тип пункта меню	Действие
Submenu (Подменю)	Раскрывается связанное с данным пунктом меню ниспадающее подменю
Procedure (Процедура)	Выполняется процедура, определенная в конструкторе меню
Command (Команда)	Выполняется команда, расположенная в поле рядом с типом пункта меню

Чтобы определить команду, которая будет выполняться при выборе пункта меню, необходимо выполнить следующие действия:

1) в столбце Result (Результат) конструктора меню выбрать из списка значение Command (Команда);

2) ввести в поле, расположенное с правой стороны списка, команду Visual FoxPro, которая будет выполняться при выборе данного пункта меню.

Обычно команды используются для вызова форм, отчетов, пользовательских процедур. Например, для запуска формы ввода списка клиентов достаточно ввести следующую команду:

DO FORM Client

В том случае, если при выборе пункта меню должна выполняться некоторая последовательность команд, вам необходимо в столбце Result (Результат) окна конструктора выбрать для пункта меню тип Procedure (Процедура) и определить требуемые команды (рисунок 18).

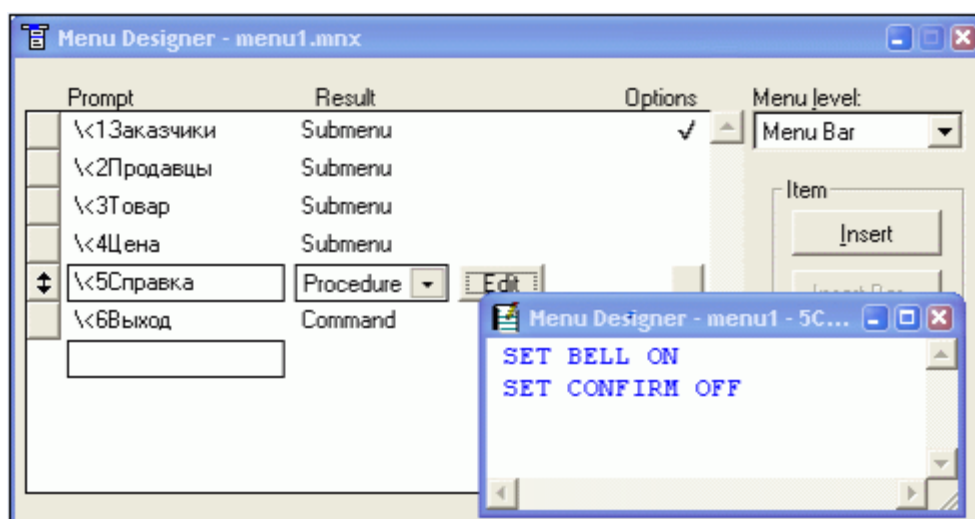


Рисунок 18 - Окно редактирования процедуры для пункта меню Справка

Для этого нажмите кнопку Create (Создать) и в открывшемся окне редактирования процедуры определите фрагмент кода, связанный с элементом строки меню. Например, для пункта меню Справка вы можете указать тип

Procedure (Процедура) и в окне создания процедуры задать имя файла справки и команду вызова справочной системы.

При вводе процедуры не используйте команду PROCEDURE, т. к. Visual FoxPro генерирует эту команду автоматически. Во время генерации для каждой процедуры, включенной в меню, Visual FoxPro создает уникальное имя и включает его вместе с указанием местонахождения процедуры в код, связанный с соответствующим элементом строки меню.

После определения текста процедуры наименование кнопки Create (Создать) изменяется на Edit (Правка).

Большинство команд меню создаваемого приложения открывает подменю. Например, пункт меню Покупатели должен содержать команды Ввод данных, Отчеты и Запросы. Для создания данного подменю выполните следующие действия:

1) Нажмите кнопку Create (Создать) пункта меню Покупатели. На экране появляется пустое окно конструктора меню. Список Menu Level (Уровень меню) нового окна содержит метку текущего пункта меню;

2) Введите в поле Prompt (Приглашение) первой строки команду Ввод данных. Используя список Result (Результат), задайте тип созданного пункта меню;

3) Во второй строке введите команду Отчеты и также задайте тип;

4) Аналогичным образом введите третью команду.

Параметры пунктов подменю определяются аналогично определению пунктов строки меню (рисунок 19). Для перехода в меню верхнего уровня используйте список Menu Level.

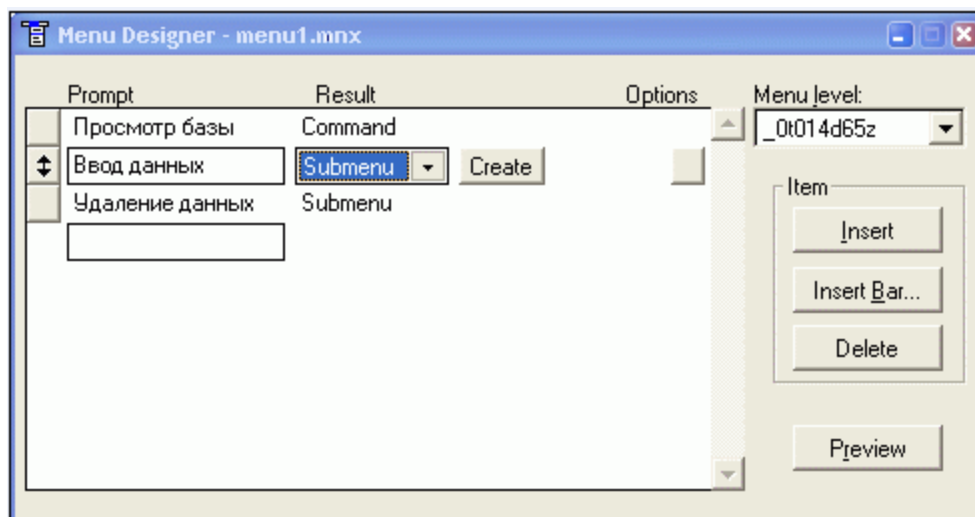


Рисунок 19 - Окно создания подменю

Для улучшения внешнего вида, а также для объединения в группы схожие по смыслу команды в меню можно использовать разделительные линии. Разделительные линии представляют собой пункт меню, в котором в поле ввода Prompt (Приглашение) вместо наименования пункта вводятся символы \-.

5.3.3 Сохранение, генерация и запуск меню

Для сохранения созданного меню выберите команду Save as (Сохранить как) в меню File (Файл). В открывшемся диалоговом окне Save as (Сохранить как) из списка Папка выберите папку, в которой вы предполагаете сохранить файл, откройте ее, в поле Save Menu (Сохранить меню) введите имя сохраняемого меню. В заключение нажмите кнопку Сохранить.

С помощью кнопки Preview (Просмотр) окна конструктора меню можно просмотреть внешний вид создаваемого меню, но нельзя его активизировать. Чтобы можно было использовать меню в приложениях, его необходимо предварительно сгенерировать. Для этого выполните следующие действия:

- 1) В меню Menu (Меню) выберите команду Generate (Генерация). Откроется диалоговое окно Generate Menu (Генерация меню) (рисунок 20);
- 2) В поле Output File (Внешний файл) введите имя файла, который будет создан в результате генерации;
- 3) Для запуска генерации описания меню нажмите кнопку Generate (Генерация).

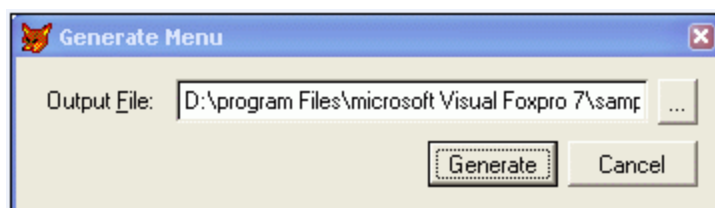


Рисунок 20 - Диалоговое окно Generate Menu

После завершения генерации можно запустить программу меню на выполнение. Для этого выполните одно из следующих действий:

- В окне проекта установите курсор на наименование созданного меню и нажмите кнопку Run (Запустить);
- В меню Program (Программа) выберите команду Do (Выполнить). В открывшемся диалоговом окне Do (Выполнить) откройте папку, в которой вы сохранили файл меню, выберите файл с расширением MPR и нажмите кнопку Do (Выполнить).

На экране появится созданное вами меню, которое заменит основное меню Visual FoxPro, если в диалоговом окне General Options (Основные параметры) перед генерацией была установлена опция Replace (Замещать).

5.3.4 Создание всплывающего меню

В Visual FoxPro имеется возможность создания всплывающего меню средствами конструктора меню.

Способ создания меню данного типа аналогичен созданию горизонтального меню в виде строки. Для этого меню, как и для обычного, можно определить оперативные клавиши и опции, устанавливаемые в диалоговом окне Prompt Options (Опции элемента меню).

Чтобы создать всплывающее меню, выполните следующую последовательность действий:

- 1) Откройте проект;
- 2) Для открытия окна конструктора меню в окне проекта перейдите на вкладку Other (Остальные) и выберите группу Menus (Меню);
- 3) Нажмите кнопку New (Новый) окна проекта;
- 4) В открывшемся диалоговом окне New Menu (Новое меню) нажмите кнопку Shortcut (Всплывающее меню). Откроется окно конструктора меню;
- 5) В поле Prompt (Приглашение) последовательно введите тексты пунктов меню и определите для них выполняемые действия (рисунок 21) ;
- 6) Для генерации выберите команду Generate (Генерация) в меню Menu (Меню);
- 7) Запустите меню на выполнение. Вид данного меню при запуске представлен на рисунке 22.

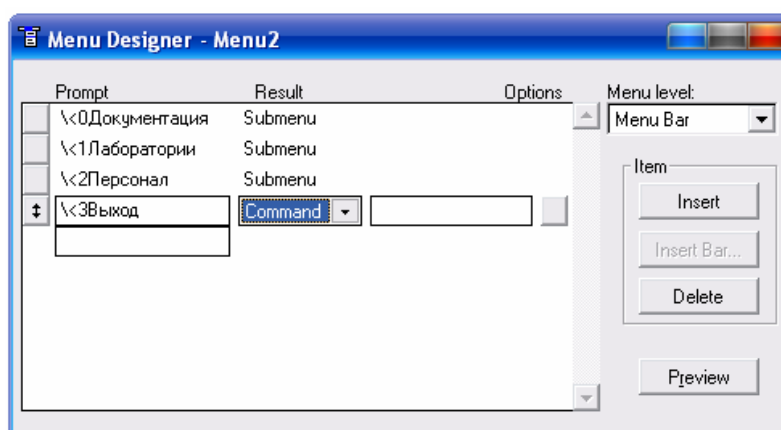


Рисунок 21- Создание меню типа Shortcut

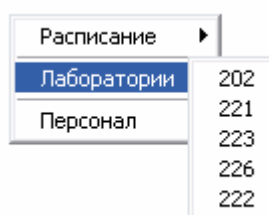


Рисунок 22 - Меню типа Shortcut

5.4 Практическая часть

5.4.1 Порядок выполнения работы

- 1) В конструкторе меню создать горизонтальное меню, согласно своего варианта;
- 2) для каждого пункта меню реализовать подменю;
- 3) назначить горячие клавиши на каждый пункт меню;
- 4) включить разделительные линии между пунктами в подменю;
- 5) при выборе одного из пунктов меню реализовать вызов формы, созданной в предыдущей работе;
- 6) определить параметры меню;
- 7) сохранить, сгенерировать и запустить меню;
- 8) создать всплывающее меню произвольного вида средствами конструктора меню;
- 9) реализовать в нем какую-либо функциональность;
- 10) сохранить, сгенерировать и запустить всплывающее меню;
- 11) сделать вывод, оформить отчет по работе.

5.4.2 Содержание отчета

- 1) тема, цель;
- 2) постановка задачи;
- 3) описание выполненной работы;
- 4) окно Menu Designer;
- 5) результат работы (меню);
- 6) содержимое Command окна;
- 7) вывод.

5.4.3 Вопросы для защиты работы

- 1) Что такое световое меню и каковы его основные элементы?
- 2) Что такое клавишное меню и каковы его основные элементы?
- 3) Какие технологии построения меню предусмотрены в Fox Pro?
- 4) Как запустить конструктор меню в Visual Fox Pro?
- 5) Какие виды меню можно создать с помощью конструктора ?
- 6) Какие существуют типы пунктов меню?
- 7) Как назначить «горячие» клавиши для пунктов меню?

6 Организация интерфейса пользователя и создание отчетов

6.1 Цель работы

Научится разрабатывать экранные формы и отчеты.

6.2 Постановка задачи

- 1) Создать экранную форму для своей предметной области
- 2) Создать отчет по данным табличных файлов

6.3 Теоретическая часть

6.3.1 Создание экранной формы с помощью Мастера форм — Form Wizard

В первой диалоговой панели Form Wizard (Step 1 — Field Selection) в раскрывающемся списке Databases надо указать имя базы данных, а в раскрывающемся списке Tables указать имя таблицы, по данным которой будет построена экранная форма. В раскрывающемся списке Available Fields следует выбрать имена полей, значения которых будут размещены в экранной форме. Перенос выбранных полей в окно Selected Fields производится с помощью кнопок-стрелок.

Во второй диалоговой панели Form Wizard (Step 2 — Style) определяется стиль исполнения экранной формы. В списке Style выбирается один из допустимых стилей: Standard, Chiseled, Shadowed, Boxed или Embossed. Результат вашего выбора можно просмотреть здесь же в специальном окне. С помощью радио-группы Button Type определяется тип отображения кнопок управления:

- Text Buttons — текстовые надписи на кнопках управления;
- Picture Buttons — пиктограммы на кнопках управления;
- No Buttons — кнопки управления отсутствуют.

В третьей диалоговой панели Form Wizard (Step 3 — Sort Order) определяется порядок сортировки данных. В списке Available Fields указываются имена полей, по значениям которых надо выполнить сортировку данных. А с помощью кнопок Ascending (по возрастанию) и Descending (по убыванию) определяется порядок сортировки. С помощью кнопки Add поля сортировки заносятся в окно Selected Fields.

В четвертой диалоговой панели Form Wizard (Step 4 — Finish) определяется заголовок созданной экранной формы, который пишется в поле ввода Type a title for your form. Далее с помощью кнопок радиогруппы выбирается вариант продолжения работы:

- Save form for later use — сохранить созданную форму;
- Save and run form — сохранить экранную форму и поставить ее на выполнение;
- Save form and modify it in the Form Designer — сохранить экранную форму и открыть ее в Конструкторе для модификации.

Если при создании таблицы, на основе которой строится экранная форма, было задано свойство Caption, то содержимое этого свойства используется в качестве надписей к полям. Если свойство Caption не задано, то для надписей к полям используются имена полей.

Созданная экранная форма готова к использованию и не требует генерации программных кодов. Для запуска созданной экранной формы надо из главного меню подать команду Program ->Do установить курсор на имя файла экранной формы и нажать кнопку Ok.

6.3.2 Создание экранной формы с помощью Конструктора форм — Form Designer

Процесс создания формы состоит в размещении компонентов формы (элементы управления, визуальные компоненты и т. д.) на листе формы и определения для них свойств и событий при выполнении различных действий. Созданная экранная форма может находиться в одном из двух состояний:

- режим проектирования — на форме размещаются компоненты, описываются их свойства и события;
- рабочий режим — файл формы поставлен на выполнение. После чего в экранную форму выводятся данные, которые можно просматривать, редактировать, удалять и производить некоторые действия (сортировка, поиск, фильтрация и т. д.) — активное состояние.

Для открытия окна Конструктора форм надо выполнить одно из действий:

1) из главного меню подать команду File -> New, в появившейся на экране диалоговой панели выбрать кнопку New File и указать тип создаваемого файла Form;

2) вызвать на экран окно проекта и выбрать вкладку Document, затем указать группу Form и нажать кнопку New;

3) вызвать на экран окно проекта и выбрать вкладку Document, затем указать группу Form и из главного меню подать команду File -> New, в появившейся на экране диалоговой панели выбрать кнопку New File и указать тип создаваемого файла Form.

При работе с Конструктором форм можно использовать четыре панели инструментов:

- Form Controls — предназначена для выбора и размещения компонентов на листе формы;
- Form Designer — предназначена для управления формой и вызовом на экран остальных трех панелей инструментов;
- Layout — предназначена для выравнивания компонентов на листе формы;
- Color Palette — предназначена для задания цвета компоненту.

Чтобы разместить компонент на форме, надо на панели инструментов выбрать нужный компонент (установить курсор мыши на нужный компонент и щелкнуть левой кнопкой). Затем курсор мыши расположить на поле формы и либо щелкнуть левой кнопкой мыши, либо, зажав левую кнопку мыши, нарисовать прямоугольный контур, внутри которого будет размещен выбранный компонент.

В каждый момент времени на форме может быть активен только один компонент. Активный компонент на форме выделен по контуру черными маркерами. Для того чтобы сделать компонент активным, надо разместить на нем курсор мыши и щелкнуть левой кнопкой мыши. Для отмены активности компонента надо либо выделить другой компонент, либо щелкнуть левой кнопкой мыши на поле формы.

Для активного компонента на экран можно вывести окно свойств Properties (окно свойств может быть постоянно открыто) с помощью команды главного меню View -> Properties. При смене активного компонента в окно Properties автоматически выводится информация для выбранного компонента. Окно Properties содержит вкладки:

- All — содержит список (в алфавитном порядке) всех свойств и Методов выбранного компонента;
- Data — содержит свойства компонента, относящиеся к источнику данных.
- Methods — содержит список всех методов компонента;
- Layout — содержит свойства, определяющие внешний вид компонента;
- Other — содержит все свойства, которые не вошли в остальные вкладки;

При работе с окном Properties выбирают нужное свойство компонента и присваивают ему значение одним из способов:

- с клавиатуры вписывают требуемое (допустимое) значение;
- с помощью кнопки открывают список, из которого выбирают нужное значение;
- с помощью кнопки выводят на экран диалоговую панель, с помощью которой формируют нужное значение.

На каждой вкладке выше списка свойств расположены три кнопки:

- 1) вывод на экран построитель выражений (формул) для задания значения свойства;
- 2) подтверждение ввода значения свойства;
- 3) отказ от введенного значения свойства.

Кроме окна Properties свойства и методы компонента можно задать программным способом.

Изменить геометрические размеры активного компонента можно, буксируя один из маркеров выделения компонента или изменяя значения свойств Height и Width.

Изменить положение компонента на форме можно, буксируя компонент или изменяя значения свойств Left и Top.

Активный компонент можно удалить либо нажав клавишу Backspace, либо нажав клавишу Del, либо из главного меню подать команду Edit -> Cut, либо с помощью кнопки графического меню

Для удобства размещения объектов на форме на листе формы выведена сетка. Установить или удалить сетку можно с помощью команды главного меню View -> Grid Line. Деления сетки (размер шага) можно задать в диалоговой панели Grid Properties, которая выводится на экран командой главного меню Format S Grid Scale.

Компоненты можно размещать строго по линиям сетки, если главного меню подать команду Format -> Snap to Grid. По умолчанию при нажатии клавиши Tab фокус управления ('порядок обхода компонентов) передается в естественном порядке слева направо и сверху вниз. Изменить порядок обхода компонентов (передачу фокуса управления) можно в одном из режимов: Interactive — интерактивный, то есть в порядке расположения элементов на форме, и в режиме By List — в естественном порядке. Для установления одного из режимов надо, находясь в Конструкторе формы, из главного меню подать команду Tools -> Options — на экране диалоговая панель Options, на которой надо выбрать вкладку Forms и в раскрывающемся списке Tab Ordering выбрать одну из команд : Interactive или By List. Команда главного меню View -> Tab Order в режиме Interactive выводит на экран порядковый номер компонента при переходе от одного компонента к другому. Удерживая клавишу Shift в зажатом состоянии, можно с помощью щелчка мыши изменить порядок передачи фокуса управления. По окончании перестановок надо нажать кнопку Reorder. Команда главного меню View -> Tab Order в режиме By List будет выводить на экран диалоговую панель Tab Order, где в окне представлены имена расположенных на форме компонент в порядке получения ими фокуса управления. С помощью мыши или кнопок By Column и By Row можно переставить имена компонент в желаемом порядке.

6.3.3 Создание отчетов с помощью мастера

Для запуска Мастера отчетов надо выполнить следующие действия.

На вкладке окна проекта выбрать кнопку Document. Если вкладка Document содержит кнопки управления, то выбрать кнопку New. Если на вкладке Document кнопок управления нет, то на экран надо вызвать контекстное меню, из которого выбрать команду New.

Эту же процедуру можно выполнить через главное меню Visual FoxPro, подав команду File, а в подменю надо указать команду New. В том и другом случае на экран выводится диалоговая панель New Report, где надо выбрать одну из кнопок:

- Report Wizard — запускает Мастер отчетов;
- New Report — запускает Конструктор отчетов.

После запуска Мастера отчетов надо определить, какие данные из одной или нескольких таблиц будут помещены в создаваемый отчет. Для этого в диалоговой панели Wizard Selection надо выбрать одну из команд:

- *One-to-Many Report Wizard* — создание отчета по данным из нескольких таблиц, тогда между таблицами будут установлены взаимосвязи «один-ко-многим»;

- *Report Wizard* — создание отчета по данным одной таблицы.

В диалоговой панели Wizard Selection надо выбрать команду Report Wizard и нажать кнопку Ok.

Все диалоговые окна Мастера отчетов содержат по пять кнопок управления:

- Cancel — отказ от построения отчета.
- Back — возврат к предыдущему шагу построения отчета;
- Next — переход к следующему шагу построения отчета;
- Finish — построить отчет по указанным данным;
- Help — выводит на экран контекстную подсказку.

На первом шаге Мастера отчетов «Step 1 — Select fields» (рисунок 23) надо определить имя таблицы, по значениям которой будет создан отчет. В раскрывающемся списке Databases and tables надо выбрать имя базы данных, которой принадлежит нужная нам таблица. Ниже раскрывающегося списка расположено окно, в котором перечислены имена всех таблиц, принадлежащих выбранной базе данных. Справа от раскрывающегося списка Databases and tables расположена кнопка обзора, при помощи которой можно определить место хранения базы данных.

В окне Available fields указаны имена полей выбранной таблицы. Поля, значения которых надо поместить в отчет, по очереди выделяют курсором и нажимают одну из кнопок.

На втором шаге «Step 2 — Group Records» (рисунок 24) создания отчета определяется необходимость группировки данных в отчете.

По умолчанию группировка данных не производится (None), то есть данные помещаются в отчет в том порядке, как они хранятся в таблице.

Для задания группировки по значению какого-либо поля надо в раскрывающемся списке «1» выбрать имя нужного поля. Допускается группировка данных внутри указанных групп данных по значениям других полей. Глубина вложения группировок — 3, то есть имеются раскрывающиеся списки с именами «2» и «3».



Рисунок 23 – Первый шаг мастера отчетов

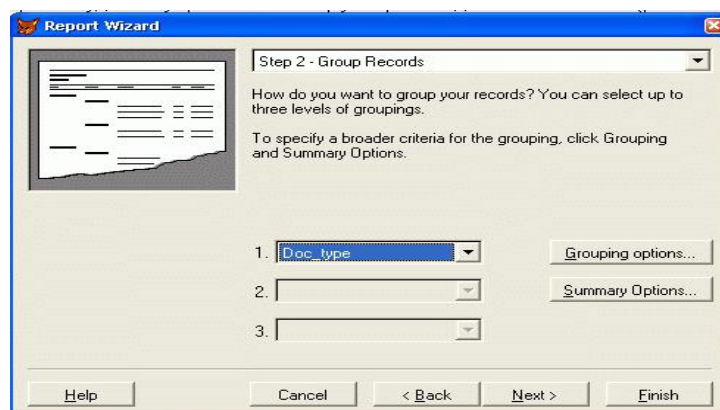


Рисунок 24 – Второй шаг мастера отчетов

Если задана группировка по значениям хотя бы одного поля, то становится доступной кнопка Grouping Options. Эта кнопка позволяет задать интервалы группировки значений поля и выводит на экран диалоговую панель Grouping Intervals (рисунок 25).

Если поле группировки символьное, то интервал группировки Entire Field задается либо по одной первой букве, либо по первым двум буквам и т. д. Если поле группировки числовое, то интервал группировки Exact Number задается либо по первому десятичному разряду, либо по первым двум десятичным разрядам и т. д. Интервал группировки может быть задан по каждому указанному полю группировки данных.

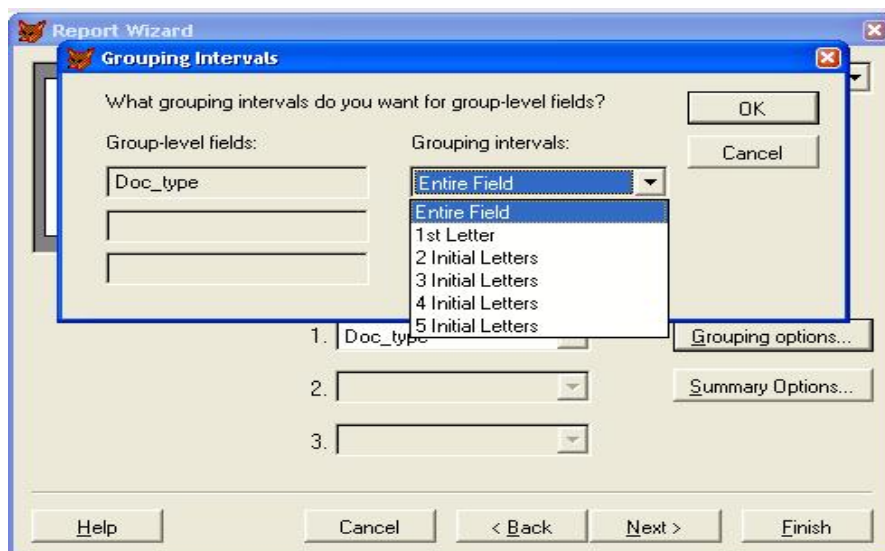


Рисунок 25 - Задание группировки данных

Для каждого числового столбца (поля) отчета можно задать вычисление итогового значения по столбцу. После нажатия кнопки Summary Options на экран выводится диалоговая панель Summary Options.

Для каждого числового столбца (поля) можно задать вычисление следующих итоговых значений:

- Sum — вычислить сумму по столбцу;
- Avg — вычислить среднее значение по столбцу;
- Count — определить количество значений в столбце;
- Min — определить минимальное значение столбца;
- Max — определить максимальное значение столбца.

С помощью кнопок радиогруппы определяется объем итоговых вычислений и область размещения данных.

Назначение кнопок:

- Detail and Summary — задает область размещения данных, вычисление промежуточных итоговых значений по группам и вычисление итогового значения по столбцу;

- Summary Only — задает область размещения данных и вычисление итогового значения по столбцу;

- No totals — задает область размещения данных.

Итоговые значения не вычисляются. Значения промежуточных итоговых значений по группам можно отображать в отчете в виде абсолютных значений, тогда флажок Calculate percent of total for sums должен быть выключен, либо в виде процента от общего итогового значения по столбцу, тогда флажок Calculate percent of total for sums должен быть включен.

На третьем шаге «Step 3 — Choose Report Style» Мастера отчетов определяется один из допустимых стилей выполнения отчета (рисунок 26).

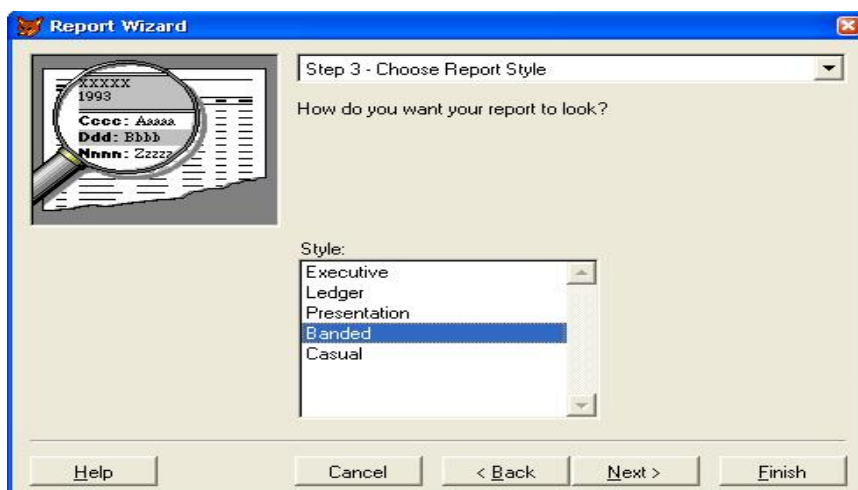


Рисунок 26 - Третий шаг мастера отчетов

Задание нужного стиля выполняется выделением соответствующей строки в окне Style.

В специальном окне, расположенном в верхнем левом углу диалоговой панели четвертого шага Мастера отчетов, можно просмотреть внешний вид выбранного стиля исполнения отчета

На четвертом шаге «Step 4 — Define Report Layout» Мастера отчетов определяется порядок размещения объектов в отчете (рисунок 27).

При создании отчета по значениям одной таблицы на этом шаге построения отчета можно определить только ориентацию листа бумаги с помощью радиогруппы Orientation, выбрав одну из кнопок:

- Portrait — вертикальное (книжное) расположение листа;
- Landscape — горизонтальное (альбомное) расположение листа.

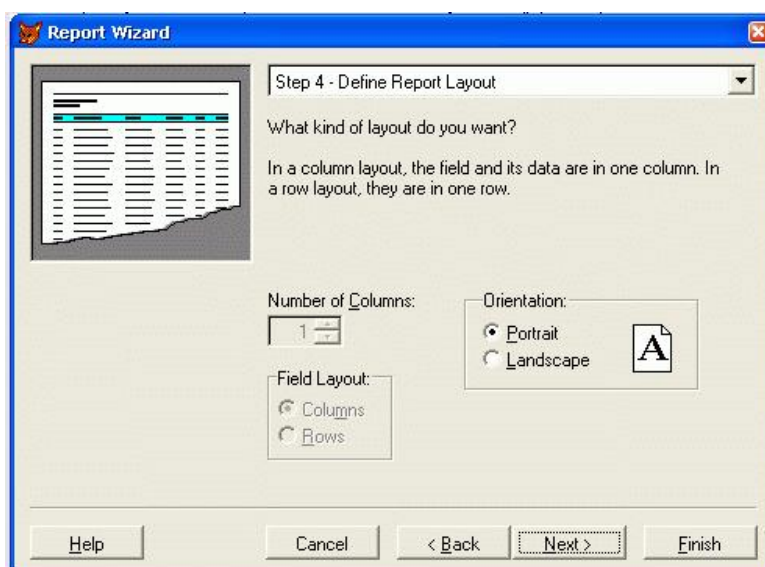


Рисунок 27 - Задание внешнего вида отчета

Остальные элементы управления, расположенные на диалоговой панели, не доступны для выбора.

На пятом шаге «Step 5 — Sort Records» Мастера отчетов определяется необходимость сортировки данных по одному или нескольким полям таблицы.

В списке Available fields of index tag указаны имена полей, по значениям которых можно выполнить сортировку данных. Если группировки полей (шаг 2) не было, то в этом списке будут представлены имена всех полей таблицы, помещаемых в отчет.

Если на втором шаге Мастера отчетов были заданы имена полей для группировки данных, то имена этих полей в список Available fields of index tag не будут включены.

Аналогично шагу 2 Мастера отчетов в списке Available fields of index tag курсором выделяется имя поля сортировки и нажатием кнопки «Add» переносится в список Selected fields.

Для удаления имени поля из списка Selected fields его надо выделить курсором и нажать кнопку Remove.

Для каждого поля сортировки можно задать направление сортировки, включив одну из кнопок радиогруппы:

- Ascending — сортировка по возрастанию значений;
- Descending — сортировка по убыванию значений.

На шестом шаге «Step 6 — Finish» Мастера отчетов задается заголовок созданного отчета и определяется дальнейший режим работы Visual FoxPro. Текст заголовка отчета пишется в поле ввода Type a title for your report.

Дальнейший режим работы с Visual FoxPro определяется включением одной из кнопок радиогруппы Select an option and click Finish:

- Save report for later use — сохранить созданный отчет;
- Save report and modify it in the Report Designer — сохранить созданный отчет и открыть его заново в Конструкторе отчетов для выполнения модификации отчета;
- Save and print report — сохранить и распечатать отчет.

На диалоговой панели размещены два флажка:

- Use display setting stored in the database — при включенном флажке для отображения значений полей используются установки, указанные в базе данных;
- Wrap fields that do not fit — при включенном флажке разрешить перенос на следующую строку символов значения полей, если значение не умещается в отведенном размере строки.

Кнопка Preview предназначена для предварительного просмотра созданного отчета перед сохранением на диске. Если созданный отчет не устраивает пользователя, то с помощью кнопки Back можно вернуться назад на один или несколько шагов и исправить отчет.

Нажатие кнопки Finish завершает создание отчета.

6.3.4 Создание отчетов с помощью конструктора

Для запуска Конструктора отчетов в диалоговой панели New Report надо выбрать кнопку New Report — экране появится рабочее окно Конструктора отчетов (рисунок 28). Как можно было заметить ранее, отчет должен содержать обязательные реквизиты: заголовок отчета, данные в табличном или произвольном виде, а также дополнительные элементы: пояснительный текст, рамки и т. д., предназначенные для удобства восприятия основных данных.

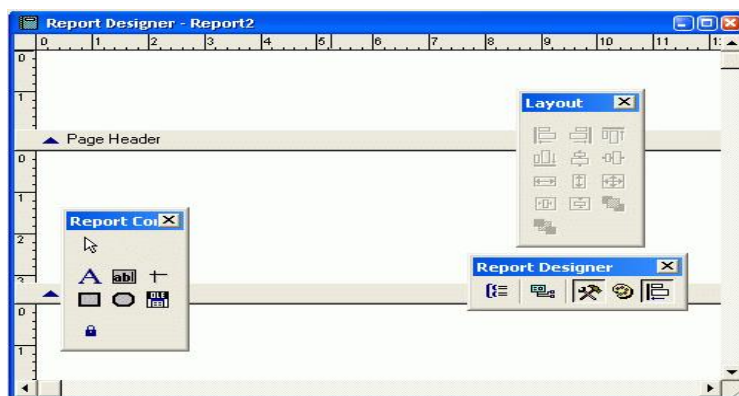










Рисунок 28 - Рабочее окно Конструктора отчетов и панель инструментов

Назначение кнопок панели инструментов:

-  I Select Object — указатель выбора объекта отчета;
-  I Label — размещает текст;
-  I Field — размещает поле;
-  I Line — рисует линию;
-  I Rectangle — рисует прямоугольник;
-  I Rounded Rectangle — рисует прямоугольник с закругленными краями;
-  I Picture/OLE Bound Control — помещает рисунок в отчет;
-  Button Lock — закрепляет выбор кнопки.

Первоначально рабочее окно Конструктора отчетов содержит три области (полосы):

Page Header — информация, размещенная в этой полосе, распечатывается в начале каждой страницы (верхний колонтитул);

Detail — размещается содержимое полей таблиц или результат вычислений над полями. Page Footer — информация, размещенная в этой полосе, распечатывается в конце каждой страницы (нижний колонтитул);

Дополнительно могут быть добавлены следующие области:
Title — информация перед основным отчетом: может быть имя отчета, сопроводительное письмо и т. д;

Group Header — информация, используемая при группировке. Группы помогают идентифицировать информацию, содержащуюся на каждом уровне

группировки;

Group Footer — итоговая информация по группе;

Summary — информация размещается один раз после основного отчета и содержит общие суммы или текст, подводящий итог содержимого отчета.

При создании и редактировании отчета возможны следующие операции:

- добавление области;
- перемещение области;
- удаление области.

Процесс создания табличного отчета состоит из следующих этапов:

- 1) определение окружения;
- 2) размещение текста;
- 3) размещение полей;
- 4) размещение линий, прямоугольников и рисунков;
- 5) перемещение объектов;
- 6) сохранение отчета.

6.4 Практическая часть

6.4.1 Порядок выполнения работы

- 1) изучите методические указания;
- 2) создайте с помощью Мастера форм экранную форму для своей предметной области;
- 3) разработанную форму перенесите в Конструктор форм и доработайте в нем;
- 4) создайте с помощью Мастера отчетов отчет по данным своих табличных файлов;
- 5) создайте с помощью Конструктора отчетов отчет по данным своих табличных файлов;
- 6) сделайте вывод о проделанной работе и оформите отчет.

6.4.2 Содержание отчета

- 1) тема, цель;
- 2) постановка задачи;
- 3) описание выполненной работы по созданию экранной формы;
- 4) разработанная экранная форма ;
- 5) описание выполненной работы по созданию отчета;
- 5) вывод.

6.4.3 Вопросы для защиты работы

- 1) Укажите назначение и основные принципы работы с Мастером форм.
- 2) Укажите назначение и основные принципы работы с Конструктором форм.
- 3) Укажите назначение режима проектирования в Конструкторе форм.
- 4) Укажите назначение панели инструментов Form Controls в Конструкторе форм.
- 5) Какие действия надо выполнить для размещения компонента на форме?
- 6) Что такое отчет?
- 7) Какие виды отчетов можно создать в Visual Fox Pro?
- 8) Назовите отличия при задании заголовка отчета при работе с Мастером и Конструктором отчетов.
- 9) Как задать таблицы при, по данным которых будет построен отчет?
- 10) Как задать несколько колонок при создании многоколоночного отчета?

7 Создание триггеров и достоверность данных

7.1 Цель работы

Научиться обеспечивать достоверность вводимых данных и работать с триггерами.

7.2 Постановка задачи

- 1) Определить условие достоверности данных на уровне поля таблицы, используя поле «field validation», вкладки «fields» и окно «Table Designer»;
- 2) Создать триггер, который после добавления или удаления записи будет выдавать соответствующее сообщение.

7.3 Теоретическая часть

7.3.1 Условия достоверности ввода данных на уровне записей

Условия достоверности позволяют контролировать ввод данных средствами сервера на уровне записей и полей таблицы. В первом случае условие определяется в окне ввода свойств таблицы, а во втором — в окне свойств поля таблицы. Проверка на уровне записи обычно используется, если необходима

проверка при добавлении или удалении записи, а также, если условие проверки изменения записи требует анализа более одного поля.

При определении условий достоверности ввода данных используются триггеры и хранимые процедуры. *Триггеры* задают действия, выполняемые при добавлении, удалении или изменении записей таблицы. *Хранимые процедуры* содержат наиболее часто используемые процедуры, выполняемые сервером базы данных. Если вы определили условия достоверности ввода данных, их проверка осуществляется независимо от способа изменения данных в таблице.

Для определения свойств таблицы откройте окно конструктора для выбранной таблицы и перейдите на вкладку **Table** (рисунок 29).

Для определения достоверности ввода данных могут использоваться триггеры добавления и изменения, а также поля ввода **Rule** (Условие) и **Message** (Сообщение) области **Record validation** (Проверка правильности ввода записей). В пол **Rule** вводится логическое выражение, которое может содержать вызов хранимой процедуры. Если значение этого выражения равно True (Истина), то считается, что введены допустимые данные и разрешается переход на другую запись или закрытие таблицы. В противном случае выводится сообщение об ошибке, которое было задано в поле **Message**.

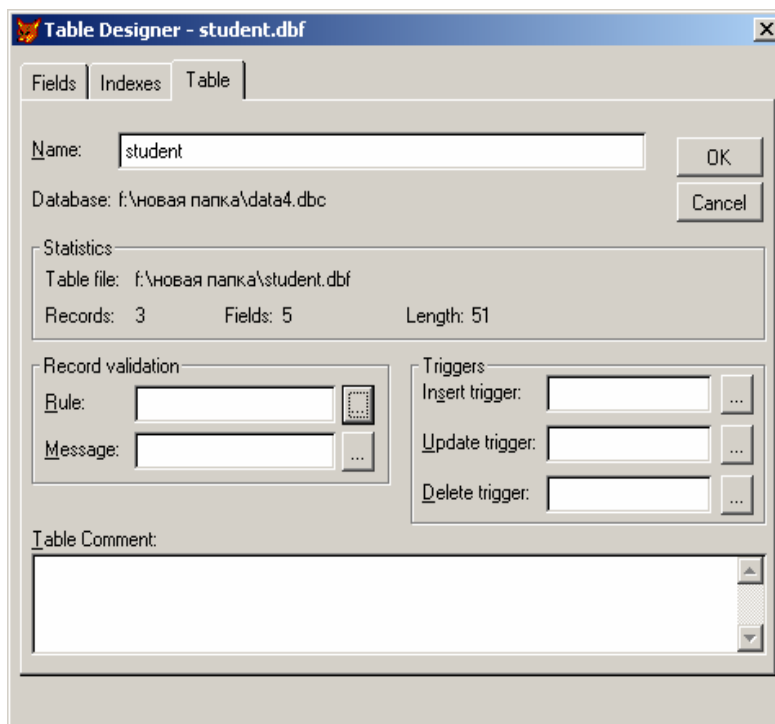


Рисунок 29 - Определение свойств таблицы

7.3.2 Триггеры

В Visual FoxPro для таблиц, входящих в состав базы данных, можно определить следующие триггеры (таблица 7).

Таблица 7 – Триггеры

Триггер	Описание
Insert	Определяет действия, которые будут выполняться после добавления новой записи в таблицу
Update	Определяет действия, которые будут выполняться после изменения записи таблицы
Delete	Определяет действия, которые будут выполняться после удаления записи из таблицы

Для определения триггеров введите в поля ввода **Insert trigger**, **Update trigger** или **Delete trigger** вкладки **Table (Таблица)** окна конструктора выбранной таблицы оператора сравнения, вызов хранимой процедуры или любое логическое выражение. Если результат вычисления выражения равен True то считается, что введены допустимые значения. В противном случае не происходит сохранения введенных данных и формируется сообщение об ошибке

При использовании хранимых процедур вы сможете не только проверить при условии достоверности ввода данных, но и задать действия, выполняемое при добавлении, удалении и изменении данных.

Вызов триггера Delete осуществляется:

- При выполнении команды DELETE;
- Когда вы помечаете запись на удаление в режиме **Browse** или **Edit**.
- Вызов триггера insert осуществляется в следующих случаях:
- При выполнении команд APPEND FROM, APPEND FROM ARRAY, APPEND BLANK;
- При добавлении в таблицу новой записи в режиме **Browse** или **Edit**;
- При выполнении команд IMPORT, INSERT - SQL, RECALL;
- При снятии метки об удалении записи в режиме Browse или Edit;
- Триггер **Update** вызывается, когда:
- Наступает любое событие, которое приводит к модификации записи (например, при изменении значения поля)
- Выполняются команды GATHER, REPLACE, REPLACE FROM ARRAY UPDATE – SQL;
- При использовании триггеров необходимо учитывать ограничения, имеющиеся в Visual FoxPro:
- При модификации записей, помеченных для удаления, и выполнении команды PACK триггеры не вызываются
- Выполнение команды ZAP не вызывает триггер Delete;
- При использовании буферизации ввода триггер update вызывается только при вызове функции TABLEUPDATE ().

При использовании триггера в окне конструктора перейдите на вкладку Table и очистите поле ввода выражения для триггера или используйте команду Delete TRIGGER.

7.4 Практическая часть

7.4.1 Порядок выполнения работы

- 1) Открыть таблицу;
- 2) В конструкторе таблицы «Table Designer» найти группу «Field validation» и установить для каждого поля свои правила достоверности данных и соответствующие сообщения об ошибках;
- 3) Во вкладке «Table» окна «Table Designer» также можно установить условия достоверности данных но только для всей таблицы;
- 4) Создать программный файл «Add.prg», который выводит предупреждение о добавлении данных в таблицу: clear; wait window 'Запись добавлена;'
- 5) Прикрепить данный программный файл к триггеру добавления во вкладке «Table»;
- 6) Создать программный файл, который выводит предупреждение об удалении данных из таблицы и закрытие таблицы: clear; wait window 'Запись удалена. Таблица закрыта;
- 7) Прикрепить данный программный файл к триггеру удаления;
- 8) Теперь при добавлении или удалении данных в таблице должен происходить вывод сообщения и проверяется достоверность данных при вводе.

7.4.2 Содержание отчета

- 1) тема, цель;
- 2) постановка задачи;
- 3) описание выполненной работы;
- 4) текст программного файла;
- 5) результат работы триггера;
- 6) вывод.

7.4.3 Вопросы для защиты работы

- 1) Что такое триггер?
- 2) Для чего предназначены триггеры?
- 3) Назовите виды триггеров.
- 4) Как создать триггер в Visual Fox Pro?
- 5) Как в Visual Fox Pro обеспечить достоверность данных?

8 Запросы к базе данных

8.1 Цель работы

Научиться создавать запросы к базе данных.

8.2 Постановка задачи

В конструкторе запросов с помощью команды Select произвести выборку данных из таблиц и поместить результат выборки в другую таблицу.

8.3 Теоретическая часть

8.3.1 Запросы к базе данных

Одним из основных назначений разработанного приложения является быстрый поиск информации в базе данных и получение ответов на разнообразные вопросы. Для этих целей в Visual FoxPro используются средства, называемые запросами.

Например, вам необходимо выбрать из таблиц информацию о клиентах, проживающих в Москве и Московской области, или сформировать список клиентов, купивших в последний месяц товаров на сумму свыше 5000 рублей, и упорядочить их в алфавитном порядке по полю, содержащему фамилии клиентов. Для решения таких задач предназначен конструктор запросов и команда SELECT языка Visual FoxPro.

С помощью конструктора запросов Visual FoxPro вы можете формировать различной сложности критерии для выбора записей из одной или нескольких таблиц, указывая при этом, какие поля должны быть отображены в запросе. Над полями, выбираемыми из таблиц с помощью запросов, можно выполнять различные вычисления.

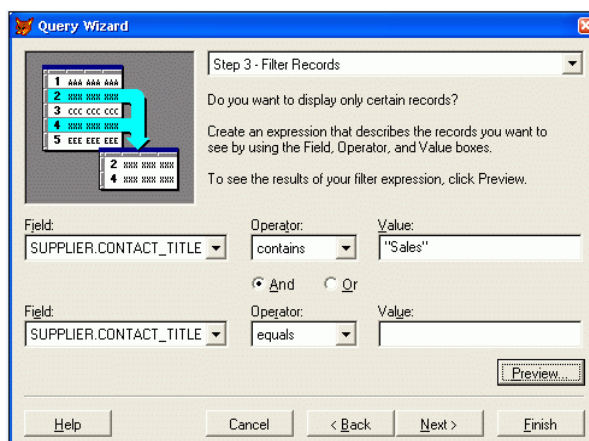


Рисунок 30 - Ввод условия выборки в мастере запросов

Результатом запроса является таблица, которую вы можете сохранить в массиве, в создаваемой новой таблице, отобразить на экране в режиме Browse (Просмотр) или вывести в виде отчета.

Для создания запросов вы можете использовать мастер запросов, который последовательно запрашивает наименования таблиц, используемых в запросе, перечень полей таблиц, критерий упорядочения и условия фильтрации данных. На рисунке 30 приведено диалоговое окно мастера, позволяющее сформировать условия фильтрации выбираемых из таблицы данных. Мы не будем рассматривать создание запросов с помощью мастера, т. к. конструктор запросов достаточно прост и работа в нем у вас не вызовет затруднений.

8.3.2 Конструктор запросов

Для создания запроса в окне конструктора запросов выполните следующие действия:

- 1) На вкладке Data (Данные) конструктора проекта выберите группу Queries (Запросы);
- 2) Нажмите кнопку New (Новый).
- 3) В открывшемся диалоговом окне New Query (Новый запрос) (рисунок 31) нажмите кнопку New Query (Новый запрос). Открывается диалоговое окно выбора таблиц Add Table or View (Добавить таблицу или представление данных) (рисунок 32);

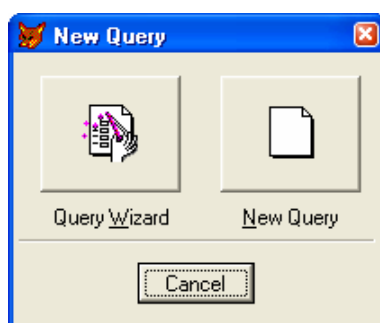


Рисунок 31 - Диалоговое окно New Query (Новый запрос)

- 4) В этом диалоговом окне выберите таблицы, данные из которых хотите использовать в запросе, и с помощью кнопки Add (Добавить) перенесите их в окно конструктора запросов;

- 5) Завершив выбор таблиц, нажмите кнопку Close (Заккрыть).

На экране появляется окно конструктора запросов (рисунок 33), которое содержит названия выбранных таблиц, а в основном меню появляется пункт Query (Запрос). Можно приступить к формированию условий запроса.

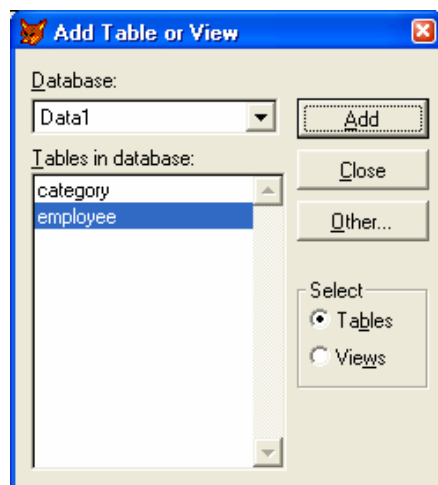


Рисунок 32 - Диалоговое окно Add Table or View (Добавить таблицу или представление данных)

Для открытия ранее созданного запроса в окне конструктора запросов на вкладке Data (Данные) окна проекта в группе Queries (Запросы) найдите модифицируемый запрос, установите на него курсор и нажмите кнопку Modify (Модифицировать).

Далее, открывая в конструкторе запросов необходимые вкладки, вы выполняете следующие действия:

- выбираете поля результирующей таблицы запроса;
- формируете вычисляемые поля;
- указываете критерии для выборки, группировки и упорядочения данных;
- задаете, куда выводить результат выборки.

В верхней части окна конструктора запросов расположена панель, на которой отображаются используемые в запросе таблицы. Ниже находятся вкладки, предназначенные для выбора полей запроса и формирования условий выборки. Назначение этих вкладок приведено в таблице 8.

Таблица 8 - Назначение вкладок окна конструктора запросов

Вкладка	Назначение
Fields (Поля)	Позволяет указать поля исходных таблиц, выбираемые в результирующий запрос
Join (Объединение)	Позволяет задать условия объединения таблиц
Filter (Фильтр)	Позволяет определить фильтры, накладываемые для выбора записей
Order By (Упорядочение)	Позволяет задать критерии упорядочения данных
Group By (Группировка)	Позволяет задать условия группировки данных
Miscellaneous (Разное)	Позволяет задать дополнительные условия, такие как признак выборки повторяющихся значений, количество или процент выбора данных

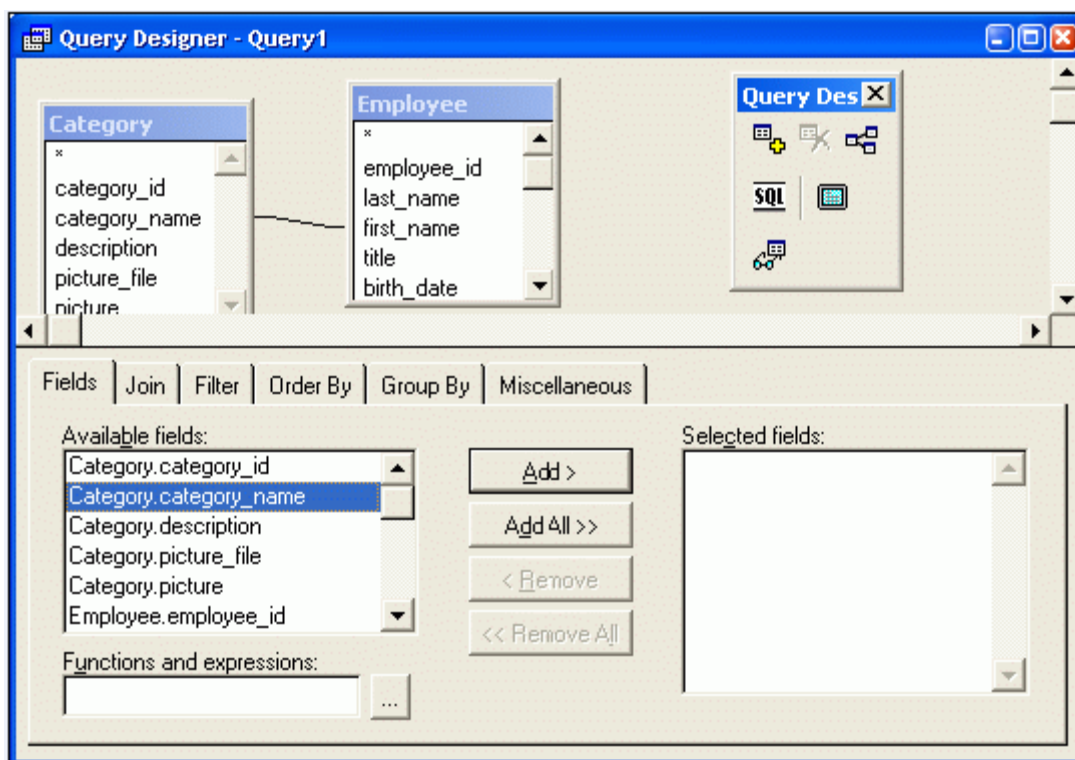


Рисунок 33 - Окно конструктора запросов с выбранной таблицей клиентов

8.3.3 Команды, используемые при формировании запросов

Для работы в окне конструктора запроса можно использовать команды меню Query и панель инструментов Query Designer (Конструктор запроса). Назначение команд и кнопок на панели инструментов описано в таблице 9.

Таблица 9 - Назначение команд меню Query и кнопок панели инструментов Query Designer

Команда меню	Назначение
Add Table (Добавить таблицу)	Добавляет в запрос новую таблицу
Remove Table (Удалить таблицу)	Удаляет выбранную таблицу из запроса
Remove Join Condition (Удалить условие объединения)	Удаляет условие объединения таблиц
Output Fields (Результирующие поля)	Открывает вкладку Fields для выбора полей результирующей таблицы
Join (Объединение)	Открывает вкладку Join для задания условия объединения таблиц

Продолжение таблицы 9

1	2
Filter (Фильтр)	Открывает вкладку Filter для задания фильтра
Order By (Упорядочение)	Открывает вкладку Order By для определения критерия упорядочения
Group By (Группировка)	Открывает вкладку Group By для определения условия группировки данных
Miscellaneous (Разное)	Открывает вкладку Miscellaneous для задания дополнительных параметров запроса
Query Destination (Результат запроса)	Открывает диалоговое окно Query Destination, в котором указывается, куда выводить результат запроса
View SQL (Показать SQL)	Открывает диалоговое окно, в котором отображается SQL-оператор, соответствующий созданному запросу
Maximize the table view (Максимизировать панель отображения)	Раскрывает панель отображения используемых в запросе таблиц на весь экран. Повторное нажатие на эту кнопку возвращает панели первоначальный размер
Add Join (Добавить условие объединения)	Открывает диалоговое окно Join Condition для задания условия объединения таблиц
Comments (Комментарии)	Открывает диалоговое окно, в котором вы можете ввести краткое описание создаваемого запроса
Run Query (Выполнить запрос)	Запускает запрос на выполнение

Для просмотра конструкции SELECT, соответствующей выборке, предназначены команда View SQL и кнопка Show the SQL window панели инструментов. Сформировав запрос, вы можете скопировать конструкцию SELECT во временный буфер Windows и использовать ее при написании программ.

8.3.4 Сохранение запроса

Сформировав в окне конструктора запросов условия выборки, вы можете их сохранить в файле с расширением QPR, что позволит сэкономить время при последующих запусках запроса.

Для сохранения условий выборки в меню File (Файл) выберите команду Save as (Сохранить как). В открывшемся диалоговом окне Save As (Сохранить

как) откройте папку, в которой хотите сохранить файл, введите в поле ввода имя файла и нажмите кнопку Сохранить.

В окне проекта запросы размещаются в разделе Queries (Запросы) вкладки Data (Данные). Для открытия запроса необходимо установить на него курсор и нажать кнопку Modify (Модифицировать) окна проекта.

8.3.5 Запуск запроса на выполнение

После задания условий для выбора записей и указания результирующих полей в окне конструктора вы можете просмотреть результаты выполнения запроса. Для этого выполните одно из следующих действий:

- нажмите кнопку Run (Выполнить) на стандартной панели инструментов;
- выберите команду контекстного меню Run Query (Выполнить запрос);
- выберите в меню Query (Запрос) команду Run Query (Выполнить запрос);
- нажмите комбинацию клавиш <Ctrl>+<Q>.

На экране появятся результаты запроса, представленные в табличном виде.

Для просмотра результатов запроса, не открытого в окне конструктора запросов, необходимо в окне проекта установить курсор на имя запроса и нажать кнопку Run (Выполнить).

8.4 Практическая часть

8.4.1 Порядок выполнения работы

- 1) изучить методические указания;
- 2) запустить конструктор запросов;
- 3) с помощью команды Select произвести выборку данных из одной или нескольких таблиц;
- 4) отсортировать отобранные данные ;
- 5) результат выборки поместить в новую таблицу;
- 7) сделать вывод о проделанной работе и оформить отчет.

8.4.2 Содержание отчета

- 1) тема, цель;
- 3) постановка задачи;
- 3) описание выполненной работы;
- 4) текст SQL-запроса;
- 5) результат SQL-запроса;
- 8) вывод.

8.4.3 Вопросы для защиты работы

- 1) Укажите область применения языка запросов SQL.
- 2) Какие операции над данными можно выполнить с помощью команд языка запросов SQL?
- 3) Напишите команду Select для выборки всех записей из таблицы
- 4) Напишите команду Select для выборки из таблицы записей, удовлетворяющих некоторому условию.
- 5) Напишите команду Select для выборки всех записей из таблицы, с указанием сортировки по возрастанию (по убыванию) по значениям какого-либо поля.
- 6) Напишите команду Select для выборки некоторых полей из разных таблиц, удовлетворяющих некоторым условиям.
- 7) Напишите команду Select для выборки всех записей из таблицы, с указанием группировки данных по значениям некоторого поля

Список использованных источников

- 1 Агальцов, В.П. Базы данных /В.П.Агальцов. –М.: Мир, 2002. – 376 с.
- 2 Попов. А.А.Создание приложений для Fox Pro 2.5/2.6 в DOS и WINDOWS/ А.А. Попов – М.: Калашников и Ко, 1997. – 660 с.
- 3 Омельченко, Л.Н. Самоучитель Visual Fox Pro 6.0 /Л.Н.Омельченко – СПб.: БХВ – Санкт-Петербург, 1999. – 512 с., ил.
- 4 Мусина, Т.В. Visual Fox Pro 7.0: учебный курс. /Т.В.Мусина, В.А.Пушенко – К.: ВЕК+; -СПб.: КОРОНА принт; -М.: Бином-Пресс, 2004. – 400 с.
- 5 Фуфаев, Э.В. Базы данных Курс лекций: учеб.пособие/ Э.В. Фуфаев, Д.Э. Фуфаев. -М.: Академия, 2005. – 320с.
- 6 Полякова, Л.Н. Основы SQL: Курс лекций: учеб.пособие /Л.Н.Полякова – М.: ИНТУИТ.РУ, Интернет-Университет Информационных Технологий, 2004. – 368 с.
- 7 Макарова, Н.В. Информатика: учебник. /Н.В.Макарова. – 3-е перераб. изд. – М.: Финансы и статистика, 2006. – 768 с.: ил.