

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Крикуненко Ирина Владимировна

Должность: директор торезского колледжа федерального государственного

бюджетного образовательного учреждения высшего образования «Донецкая

академия управления и государственной службы»

Дата подписания: 14.04.2025 12:17:11

Уникальный программный ключ:

d849e6db1fe707f55c973710998089a95

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

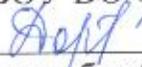
**Торезский колледж (филиал)**

федерального государственного бюджетного образовательного учреждения  
высшего образования

«Донецкая академия управления и государственной службы»  
(Торезский колледж ФГБОУ ВО «ДОНАУИГС»)

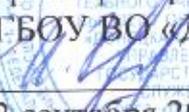
**СОГЛАСОВАНО**

Врио заместителя директора  
ФГБОУ ВО «ДОНАУИГС»

 О.В. Дорожжина  
02 сентября 2024 г.

**УТВЕРЖДАЮ**

Директор Торезского колледжа  
ФГБОУ ВО «ДОНАУИГС»

 И.В. Крикуненко  
02 сентября 2024 г.



**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ  
ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ И ПРАКТИЧЕСКИХ ЗАНЯТИЙ  
ПМ 11 РАЗРАБОТКА, АДМИНИСТРИРОВАНИЕ И ЗАЩИТА БАЗ  
ДАНЫХ**

основной профессиональной программы  
среднего профессионального образования по специальности  
**09.02.07 Информационные системы и программирование**  
**Квалификация: Программист**  
(программа подготовки специалистов среднего звена)

г. Торез  
2024

## ПЕРЕЧЕНЬ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

№	№, наименование раздела (темы)	Наименование практического занятия
1	Тема 1.1 Основы хранения и обработки данных. Проектирование БД.	Практическая работа №1 «Сбор и анализ информации»
2	Тема 1.1 Основы хранения и обработки данных. Проектирование БД.	Практическая работа №2 «Проектирование реляционной схемы базы данных в среде СУБД»
3	Тема 1.1 Основы хранения и обработки данных. Проектирование БД.	Практическая работа №3 «Приведение БД к нормальной форме 3НФ»
4	Тема 1.2 Разработка и администрирование БД	Практическая работа №4 «Создание базы данных в среде разработки»
5	Тема 1.2 Разработка и администрирование БД	Практическая работа №5 «Организация локальной сети. Настройка локальной сети»
6	Тема 1.2 Разработка и администрирование БД	Практическая работа №6 «Установка и настройка SQL-сервера»
7	Тема 1.2 Разработка и администрирование БД	Практическая работа №7 «Экспорт данных базы в документы пользователя»
8	Тема 1.2 Разработка и администрирование БД	Практическая работа №8 «Импорт данных пользователя в базу данных»
9	Тема 1.2 Разработка и администрирование БД	Практическая работа №9 «Выполнение настроек для автоматизации обслуживания базы данных»
10	Тема 1.2 Разработка и администрирование БД	Практическая работа №10 «Мониторинг работы сервера»
11	Т.1.2. Системы управления базами данных (СУБД) и манипулирование данными	Практическая работа №11 «Выполнение резервного копирования»
12	Т.1.2. Системы управления базами данных (СУБД) и манипулирование данными	Практическая работа №12 «Восстановление базы данных из резервной копии»
13	Т.2.1. Проектирование БД и создание таблиц	Практическая работа №13 «Реализация доступа пользователей к базе данных»
14	Т.2.2. Индексирование таблиц	Практическая работа №14 «Мониторинг безопасности работы с базами данных»
15	Т.2.3. Сортировка, поиск, фильтрация	Практическая работа №16 «Установка приоритетов»
16	Т.2.4. Взаимосвязи таблиц	Практическая работа №17 «Развертывание контроллеров домена»

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Методические рекомендации по выполнению практических работ ПМ.11 Разработка, администрирование и защита баз данных предназначены для реализации программы по специальности **09.02.07 Информационные системы и программирование**, входящей в укрупнённую группу профессий 09.00.00 Информатика и вычислительная техника.

Задания на практические работы разработаны и составлены на основе рабочей программы ПМ.11 Разработка, администрирование и защита баз данных. Указанный профессиональный модуль относится к профессиональному циклу в структуре основной профессиональной образовательной программы.

Методические рекомендации предназначены для подготовки и выполнения обучающимися практических работ по профессиональному модулю. Методические указания по выполнению практических работ ПМ.11 Разработка, администрирование и защита баз данных включают в себя теоретические сведения, практические задания, контрольные вопросы для каждой практической работы.

Выполнение практических работ предназначено для получения умений и навыков работы с СУБД MS Access, Visual FoxPro.

Наличие практических работ позволяет отследить уровень умений и степень подготовленности обучающихся к выполнению профессиональных задач.

#### Практическая работа №1 «Сбор и анализ информации»

##### Тема 1.1 Основы хранения и обработки данных. Проектирование БД.

**Цель работы:** «Знакомство реляционной БД, освоение принципов проектирования БД»  
**Материально-техническое обеспечение:** Компьютер, операционная система Windows 7, MS Access 2010

##### Краткие теоретические сведения:

###### Общие сведения о программе MS Access

Access является наиболее сложной программой из всех офисных приложений Microsoft Office. Чтобы начать работу с этой программой, вначале необходимо создать структуру базы данных, подробно ее описать, а затем создать различные формы.

ACCESS – это реляционная СУБД. Это означает, что с ее помощью можно работать одновременно с несколькими таблицами базы данных, эти таблицы между собой связаны. Таблицу ACCESS можно связать с данными, хранящимися на другом компьютере. Данные ACCESS очень просто комбинировать с данными EXCEL, WORD и другими программами Office.

Access во многом похож на Excel. Основное различие между таблицей БД и электронной таблицей – в системе адресации: в электронной таблице адресуется каждая ячейка, а в таблице БД – только поля текущей записи.

###### **База данных состоит из следующих компонентов:**

**Таблицы** – основные объекты базы данных (БД). В них хранятся данные. Реляционная база данных может иметь много взаимосвязанных таблиц. Сведения по разным вопросам следует хранить в разных таблицах. В БД столбцы называются **полями**, а строки – **записями**. Для работы таблицу необходимо открыть. Перед окончанием работы ее следует закрыть, предварительно сохранив все изменения.

**Запрос** – это средство, с помощью которого извлекается из базы данных информация, отвечающая определенным критериям. Результаты запроса представляются в виде записи из таблицы, а только те, которые удовлетворяют запросу.

**Формы** – обеспечивают более наглядную работу с таблицами, с помощью форм в базу вводят новые данные или просматривают имеющиеся.

**Отчеты** – средство представления данных таблиц. Отчеты могут быть оформлены надлежащим образом и распечатаны в том виде, в котором требуется пользователь.

**Макросы** – набор из одной или более макрокоманд, выполняющих определенные операции (открытие форм, печать отчетов)

**Модули** - это программы, написанные на языке программирования Visual Basic.

###### Таблицы

С таблицами можно работать в двух режимах — **таблицы и конструктора**.

**В режиме таблицы** можно начать ввод данных сразу, структура таблицы при этом будет создаваться автоматически. Полям присваиваются имена с последовательными номерами («Поле1», «Поле2» и т. д.).

На вкладке **Создание** в группе **Таблицы** щелкните **Таблица**.

Приложение Access создаст базу данных с пустой таблицей с именем «Таблица1» и откроет эту таблицу в режиме таблицы. При этом будет выделена первая пустая ячейка столбца **Добавить поле**.

1. Чтобы добавить данные, начните их вводить в первую пустую ячейку или вставьте данные из другого
2. Для переименования столбца (поля) дважды щелкните заголовок столбца и введите новое имя. Рекомендуется присваивать полям значимые имена, чтобы при просмотре области **списка полей** было понятно, что содержится в каждом поле.
3. Для перемещения столбца сначала щелкните его заголовок, чтобы выделить столбец, а затем перетащите столбец в нужное место.

4. Кроме того, можно выбрать сразу несколько смежных столбцов, а затем одновременно перетащить их в новое место. Чтобы выбрать несколько последовательно расположенных столбцов, щелкните заголовок первого столбца, а затем, удерживая нажатой клавишу SHIFT, щелкните заголовок последнего столбца.

###### Создание таблицы в режиме конструктора

В режиме конструктора сначала создается структура новой таблицы. Затем можно переключиться в режим таблицы для ввода данных или ввести данные, используя другой метод, например, вставку или импорт.

1. На вкладке **Создание** в группе **Таблицы** щелкните **Конструктор таблиц**.
2. Для каждого поля в таблице введите имя в столбце **Имя поля**, а затем в списке **Тип данных** выберите тип данных.
3. Когда все необходимые поля будут добавлены, сохраните таблицу:

4 Щелкните значок **Кнопка Microsoft Office**, а затем выберите команду **Сохранить**.

###### Создание таблицы на основе шаблона

В приложении Access имеются шаблоны для часто используемых типов таблиц. Одним щелчком мыши можно создать полную, сконфигурированную и готовую к использованию структуру таблицы. Чтобы привести таблицу в соответствие со своими запросами пользователь может добавлять или удалять поля.

1. На вкладке **Создание** в группе **Таблицы** щелкните **Шаблоны таблицы** и затем выберите из списка один из доступных шаблонов.

2. Данные можно ввести в первую пустую ячейку таблицы или вставить их из другого источника в соответствии с инструкцией.

###### Удаление столбца

Щелкните заголовок столбца правой кнопкой мыши, а затем выберите команду

**Удалить столбец**

###### Свойства полей. Типы полей

**Поля** – это основные элементы структуры базы данных, которые обладают свойствами. Основным свойством любого поля является его длина, которая выражается в символах. Уникальным свойством любого поля является его **Имя**.

Все записи состоят из полей, в каждом из которых содержится отдельный фрагмент информации. При создании таблицы необходимо сначала указать имена и типы полей, а затем вводить данные

Разные типы полей имеют разное назначение и разные свойства

Типы данных	Описание
<b>Текстовый</b>	Произвольный набор(не более 255) букв, цифр, знаков пунктуации. Может использоваться для хранения имен, адресов, телефонов, кратких описаний.
<b>Числовой</b>	Используется для хранения чисел
<b>Дата/Время</b>	Для хранения информации о дате и времени до 9999 года включительно
<b>Денежный</b>	Денежные значения и числовые данные, используемые в математических расчетах
<b>Поле MEMO</b>	Для хранения комментариев до (64Кб)
<b>Счетчик</b>	Специальное числовое поле, в котором Access автоматически присваивает уникальный порядковый номер каждой записи.. Значение полей типа счетчика обновлять нельзя
<b>Поле объекта OLE</b>	Позволяет поместить в таблицу произвольный объект из другой программы
<b>Гиперссылка</b>	Адрес в Интернете, электронный адрес или адрес файла

<b>Мастер подстановок</b>	Создает поле, в котором предлагается выбор значений из списка или из поля со списком, содержащего набор постоянных значений или значений из другой таблицы
<b>Логический</b>	Содержит одно из двух значений: Истина или ЛОЖЬ

#### Правильная структура базы данных

В основе процесса создания базы данных лежат определенные принципы. Первый принцип заключается в необходимости исключать повторяющиеся (или лишние) данные.

Второй принцип касается важной роли правильных и полных данных. Если база данных содержит неправильные данные, все отчеты, в которых данные объединяются, будут также содержать неверные сведения. Это может привести к принятию неправильных решений на основе отчетов.

#### Процесс разработки

Процесс разработки базы данных включает следующие шаги: Определение цели создания базы данных, поиск и организация необходимых данных, распределение данных по таблицам, преобразование элементов данных в столбцы

#### Задание первичных ключей.

Каждая таблица должна содержать столбец или набор столбцов для однозначного определения каждой строки таблицы. Как правило, в этих целях используется уникальный идентификационный номер, например код сотрудника или серийный номер. В базе данных такие сведения носят название **первичного ключа** таблицы.

#### Создание связей между таблицами

Проанализируйте все таблицы и определите, как данные одной таблицы связаны с данными других таблиц. Добавьте в таблицы поля или создайте новые таблицы для создания необходимых связей.

#### Порядок выполнения практической работы:

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

#### Задания для выполнения практической работы:

##### СОЗДАНИЕ БАЗОВЫХ ТАБЛИЦ

Руководитель малого предприятия, выполняющего сборку персональных компьютеров из готовых компонентов, заказал разработку базы данных, основанной на двух таблицах комплектующих. Одна таблица содержит данные, которые могут отображаться для клиентов при согласовании спецификации изделия, - в ней указаны розничные цены на компоненты. Вторая таблица предназначена для анализа результатов деятельности предприятия - в ней содержатся оптовые цены на компоненты и краткая информация о поставщиках (клиенты предприятия не имеют доступа к данным этой таблицы):

1. Запустите программу Microsoft Access (Пуск → Программы → Microsoft Access).
2. В окне Microsoft Access включите переключатель Новая база данных и щелкните на кнопке ОК.
3. В окне *Файл новой базы данных* выберите свой диск U, создайте новую папку с именем *База данных* и в нее сохраните файл, дав ему имя: *Комплектующие*. Убедитесь, что в качестве типа файла выбрана база данных Microsoft Access, и щелкните на кнопке *Создать*. Откроется окно новой базы - *Комплектующие: база данных*.

4. Откройте панель *Таблицы*.

5. Дважды щелкните на значке *Создание таблицы* в режиме конструктора - откроется бланк создания структуры таблицы.

6. Для первой таблицы введите следующие поля:

Комплектующие : таблица		Тип данных	Описание
Компонент	Имя поля	Текстовый	
Модель	Модель	Текстовый	
Основной параметр	Основной параметр	Текстовый	
Цена	Цена	Денежный	

7. Щелкните на поле *Цена*. В нижней части бланка задайте свойство *Число десятичных знаков* равным 2.

8. Для связи с будущей таблицей поставщиков надо задать ключевое поле. Поскольку здесь ни одно поле явно не претендует на "уникальность", используем комбинацию полей *Компонент* и *Модель*. Выделите оба поля в верхней части бланка (при нажатой клавише *SHIFT*). Щелчком правой кнопки мыши откройте контекстное меню и выберите в нем пункт *Ключевое поле*.

9. Закройте окно *Конструктора*. При закрытии окна дайте таблице имя *Комплектующие*.

10. Повторив действия пп. 5...9, создайте таблицу *Поставщики*, в которую войдут следующие поля.

Поставщики : таблица		Тип данных
Компонент	Имя поля	Текстовый
Модель	Модель	Текстовый
Цена оптовая	Цена оптовая	Денежный
Поставщик	Поставщик	Текстовый
Телефон	Телефон	Текстовый
Адрес	Адрес	Текстовый
Примечание	Примечание	Поле MEMO

Обратите внимание на то, что поле номера телефона является текстовым, несмотря на то, что обычно номера телефонов записывают цифрами. Это связано с тем, что они не имеют числового содержания. Номера телефонов не сравнивают по величине, не вычитают из одного номера другой и т.д. Это типичное текстовое поле.

! Не забудьте задать ключевые поля в таблице «Поставщики» - это поле «Компонент» и поле «Модель»

#### СОЗДАНИЕ МЕЖТАБЛИЧНЫХ СВЯЗЕЙ

1. В окне *Комплектующие: база данных* откройте панель *Таблицы*. Убедитесь, что на ней присутствуют значки ранее созданных таблиц *Комплектующие* и *Поставщики*.

2. Разыщите на панели инструментов кнопку *Схема данных*. Если есть сложности, найдите команду строки меню: *Сервис* → *Схема данных*. Воспользуйтесь любым из этих средств, чтобы открыть окно *Схема данных*. Одновременно с открытием этого окна

открывается диалоговое окно *Добавление таблицы*, на вкладке *Таблицы* которого можно выбрать таблицу, между которыми создаются связи.

3. Щелчком на кнопке *Добавить* выберите таблицу *Комплектующие* и *Поставщики*, в окне *Схема данных* откроются списки полей этих таблиц.

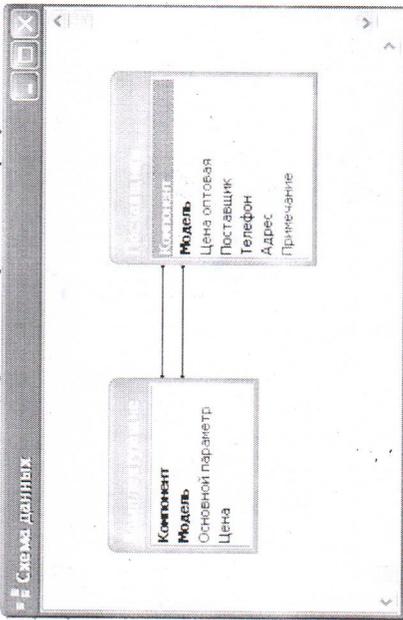
4. При нажатой клавише *SHIFT* выделите в таблице *Комплектующие* два поля - *Компонент* и *Модель*.

5. Перетащите эти поля на список полей таблицы *Поставщики*. При отпускании кнопки мыши автоматически откроется диалоговое окно *Схема связей*.

6. На правой панели окна *Изменение связей* выберите поля *Компонент* и *Модель* таблицы *Поставщики*, включаемые в связь. Не устанавливайте флажок *Обеспечение целостности данных* - в данном упражнении это не требуется, но может препятствовать постановке учебных опытов с таблицами.

9. Закройте диалоговое окно *Изменение связей* и в окне *Схема данных* рассмотрите образовавшуюся связь. Убедитесь в том, что линией связи можно выделить щелчком левой кнопки мыши, а щелчком правой кнопки мыши откроется контекстное меню, позволяющее разрывать связь или редактировать ее.

Результат выполнения упражнения приведен на рисунке.



### СОЗДАНИЕ ЭКРАННОЙ ФОРМЫ

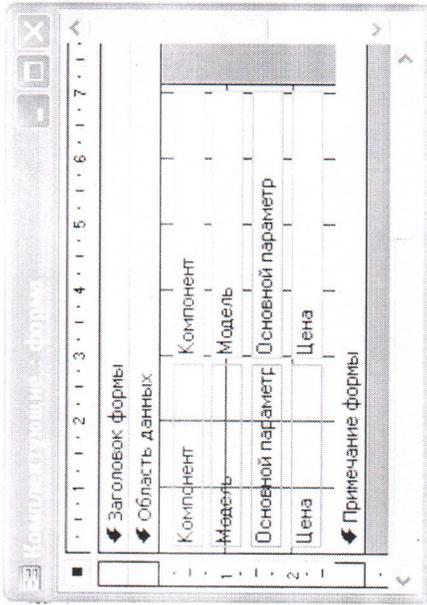
Наполнить базу данных можно двумя способами:

- Непосредственно вводя информацию в таблицы БД;
- Вводя информацию через специально созданные экранные формы.

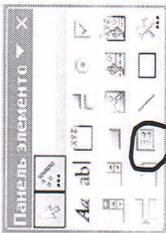
Экранная форма должна обеспечивать однократный ввод данных во взаимосвязанные таблицы.

Создайте с помощью мастера многотабличную форму, состоящую из двух форм (главной и подчиненной). В качестве источника данных для основной части формы укажите таблицу *Комплектующие*, для подчиненной формы - таблицу *Поставщики*. Таблица *Комплектующие* является главной по отношению к таблице *Поставщики*.

1. Создайте форму в режиме мастера на основе полей таблицы *Комплектующие*.
2. Затем откройте созданную форму в режиме конструктора

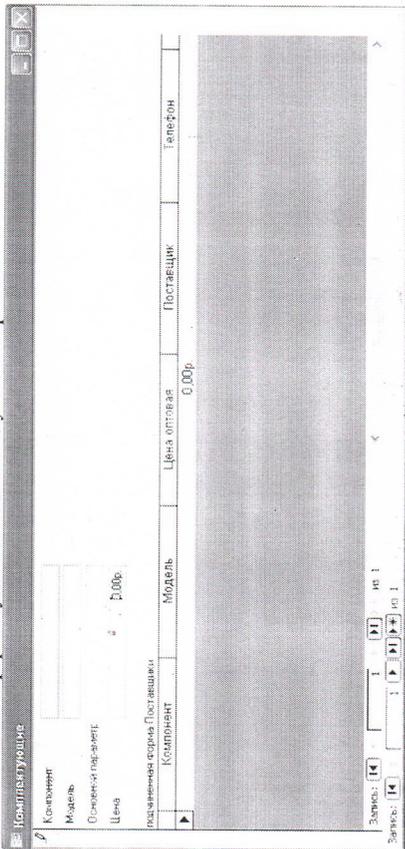


На панели элементов найдите кнопку *Подчиненная форма/отчет*



Нажав кнопку щелкните левой кнопкой мыши по *Примечанию формы Комплектующие* и создайте подчиненную форму на основе имеющейся таблицы *Поставщики*.

Созданная форма будет выглядеть следующим образом:



Данные для заполнения таблицы *Комплектующие*.

Компонент	Модель	Основной параметр	Цена
Винчестер HDD	Seagate ST902503FJD105-RK	250 Gb	5903р.
Винчестер HDD	Hitachi SATA-II HDT721016SLA380	160 Gb	1618р.
Винчестер HDD	Maxtor SATA-II STM3160813AS	160 Gb	1624р.

Практическая работа №2 «Проектирование реляционной схемы базы данных в среде СУБД».

**Тема 1.1 Основы хранения и обработки данных. Проектирование БД.**

**Цель работы:** «Преобразование реляционной БД»

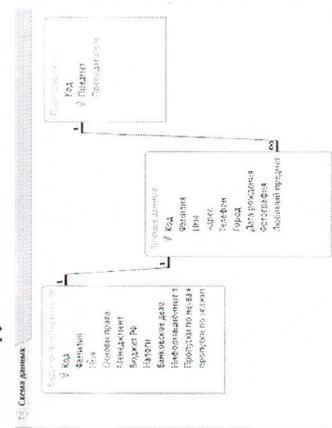
**Материально-техническое обеспечение:** Компьютер, операционная система Windows 7, MS Access 2010

**Краткие теоретические сведения:**

**Связи между таблицами**

Для полноценной работы Базы данных нужно наладить связи между таблицами, сделать их зависимыми друг от друга, чтобы по данным одной таблицы можно было узнать данные другой таблицы.

Связь между таблицами осуществляется с помощью числового идентификатора. Взяв идентификатор из одной таблицы, можно найти данные в другой таблице. Для установки связи между двумя таблицами необходимо в группе. **Показать или скрыть**, расположенной на вкладке **Работа с базами данных**, выбрать команду **Схема данных**. Появится окно **Схема данных**, в котором отображаются зависимости между таблицами.



Добавить новую связь можно, дважды щелкнув левой кнопкой мыши по окну или выбрав соответствующую команду на вкладке **Работа с базами данных** → **Схема данных** → **Конструктор** → **Изменить связь**. Появится окно **Изменение связей**, в котором нужно указать таблицы между которыми устанавливаются связи.

Как же влияет установление связи в таблицах на работу Microsoft Access?

Программ контролирует ввод данных в связанные таблицы по следующим правилам:

- запрещено добавлять значение, отсутствующее в исходной таблице, в связанную таблицу. То есть, запрещен ввод нового любого предмета в таблицу **Личные данные**, если в таблице **Преподаватели** этот предмет отсутствует;
- запрещено удаление в исходной таблице записей, на которые есть ссылка в связанной таблице;
- при изменении записей в исходной таблице меняются записи в связанной таблице.

Связанные записи можно удалять только через исходную таблицу.

**Порядок выполнения практической работы:**

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;

Материнская плата	Socket AM2+ ASUS	M3N78 GF8200 DDR2 SVGA PCI-Ex16 Sound GLAN ATX	3058р.
Материнская плата	Socket AM2 Gigabyte	GA-M56S-S3 nForce560 DDR2 PCI-E 8ch GLAN 1394 ATX	2556р.
Материнская плата	Socket AM2 MSI	K9N NEO-F nVidia nForce 550 DDR2 SATA-II Audio, GLAN ATX	2313р.
Корпус	Midtower Foxconn TH-002 Silver-Black 500W ATX (P4) гориз.БП Gigabyte	GZ-X5BPD black w/o PSU ATX	4027р.
Корпус	Midtower GMC	C-50* 350W Black, SlimATX 2.0 (serial-ATA, PCI-E, fan 80mm) 4xUSB2.0	2552
Видеокарта	PCI-E 512 Mb ASUS	E4N4650DI DDR2 2DVI TV	2940
Видеокарта	PCI-E 512 Mb Sapphire	HD4830 DDR3 Dual DVI	4735
Видеокарта	PCI-E 1 Gb MSI	N9600GT-T2DIG-OC GF9600GT DDR3 TV DVI	5687
Процессор	Pentium Dual-Core	E5400 (2.7 GHz 2Mb 800 MHz EM64T) Socket 775	3995
Процессор	Celeron Dual Core	E1200 (1.6 GTu/ 512Kb 800MHz Socket 775 / EM64T)	1874
Процессор	Athlon 64 X2 Dual Core	4800+ Socket AM2 AMD	2121
Накопитель DVD±RW	NEC	AD-7201A Silver/DVD±RW/DL ±12x/DL/RW +2.4 -2x/DVD-RAM 12x/±20x/W/+8x-6x RW	1085
Накопитель DVD±RW	ASUS	DRW-20B1LT SATA Silver DVD 20x/6x/8x/16x, CD 48x/32x/48x	1241
Накопитель DVD±RW	NEC	AD-7200A IDE Silver 8x/DL/20xDVD±R/8xDVD+R/6xDVD-RW/5xRAM/16xDVD-ROM	1015
Монитор	ViewSonic VA903B*	TFT 19"	5951
Монитор	Benq G900AD	TFT 19"	5951
Монитор	Samsung 2023NW	TFT 20"	6390
Клавиатура	BTC	5213 USB и-нет, м-медиа, эргономичная	385
Клавиатура	Genius KB06X	с подставкой под запясть (Palm Rest) black PS/2	303
Клавиатура	Genius KB06XE	Black PS/2	232
Мышь	BTC M843U-Grey	оптика, 1000 dpi, серая, USB	249
Мышь	Genius Traveler 355 Laser	оптический скролл, mini, лазерная, 1600/800dpi, USB, black	834
Мышь	Genius NetScroll 100 Optica	(Знч и scroll) PS/2 silver	156

Данные для заполнения информации о поставщиках придумайте свои, цены оптовые сделайте ниже цен реализации.

**Контрольные вопросы:**

1. Что такое СУБД?
2. Что такое таблица в БД?
3. Назовите основные элементы БД в MS Access.

- выводы.

4. Выполненную работу и отчет по проделанной работе представлять преподавателю.

#### **Задания для выполнения практической работы:**

##### **СОЗДАНИЕ ЗАПРОСА НА ВЫБОРКУ**

В этом упражнении мы создадим запрос на выборку жестких дисков, имевших емкость не менее 8 Гбайт при цене менее 150 условных единиц. Результирующая таблица должна содержать также адрес поставщика и номер его телефона.

1. Запустите программу Microsoft Access-2010 (Пуск → Программы → Microsoft Access).
2. В окне Microsoft Access включите переключатель *Открыть базу данных*, выберите ранее созданную базу *Комплекты* и щелкните на кнопке *ОК*.
3. В окне *Комплекты*: база данных откройте панель *Запросы*. Дважды щелкните на значке *Создание запроса* в режиме *Конструктора* - откроется бланк запроса по образцу. Одновременно с ним откроется диалоговое окно *Добавление таблицы*.
4. В окне *Добавление таблицы* выберите таблицу *Поставщики* и щелкните на кнопке *Добавить*. Закройте окно *Добавление таблицы*.

5. В списке полей таблицы *Поставщики* выберите поля, включаемые в результирующую таблицу: *Компонент*, *Модель*, *Цена отпоявая*, *Поставщик*, *Телефон*. Выбор производите двойными щелчками на именах полей.

6. Задайте условие отбора для поля *Компонент*, в соответствующую строку введите *Жесткий диск*. Из таблицы будут выбираться не все изделия, а только жесткие диски.

7. Задайте условие отбора для поля *Цена отпоявая*. В соответствующую строку введите < 150. Из таблицы будут выбираться только изделия, имеющие цену менее 150 условных единиц.

8. Нам еще надо задать условие отбора по основному потребительскому параметру - емкости жесткого диска. Однако в таблице *Поставщики* такого поля нет. С другой стороны, в ней есть поле *Модель*, которое однозначно определяет параметры изделия. Благодаря тому, что по полю *Модель* у нас установлена связь с таблицей *Комплекты*, мы получаем возможность ввести в запрос поле *Основной параметр*, взяв его из другой таблицы.

Добавьте список полей таблицы *Комплекты* в верхнюю часть бланка запроса по образцу. Для этого щелкните правой кнопкой мыши в верхней области бланка и в открывшемся контекстном меню выберите пункт *Добавить таблицу* - откроется уже знакомое нам окно *Добавление таблицы*. Выберите в нем таблицу *Комплекты*.

9. Двойным щелчком на поле *Основной параметр* в списке полей таблицы *Комплекты* введите это поле в бланк запроса по образцу.

10. В строке *Условие отбора* столбца *Основной параметр* введите условие >8 (емкость диска - более 8 Гбайт).

11. Закройте бланк запроса по образцу. При закрытии запроса введите его имя - *Выбор комплектов*.

12. В окне *Комплекты*: база данных откройте только что созданный запрос и расмотрите результирующую таблицу. Ее содержательность зависит от того, что было введено в таблицы *Комплекты* и *Поставщики* при их наполнении в упражнении 1. Если ни одно изделие не соответствует условию отбора и получившаяся результирующая таблица не имеет данных, откройте базовые таблицы и заполните их модельными данными, позволяющими проверить работу запроса.

13. По окончании исследования откройте все открытые объекты и завершите работу с программой Microsoft Access.

##### **СОЗДАНИЕ ЗАПРОСОВ "С ПАРАМЕТРОМ"**

Выше мы рассмотрели, как действует условие отбора, но должны отметить его существенный недостаток. Пользователь базы данных работает с запросами, которые ему подготовил разработчик. Если, например, разработчик предусмотрел запрос, отбирающий

изделия, имеющие цену менее 150 условных единиц, то пользователь базы уже не в состоянии отобразить изделия, цена которых менее 140 условных единиц, поскольку у него нет соответствующего запроса.

Специальный тип запросов, называемый запросами "с параметром", позволяет пользователю самому ввести критерий отбора данных на этапе запуска запроса. Этот прием обеспечивает гибкость работы с базой.

Создадим простой запрос, позволяющий отбирать процессоры, предельную цену которых пользователь может задать сам при запуске запроса.

1. Запустите программу Microsoft Access - 2010 (Пуск → Программы → Microsoft Access).

2. В окне Microsoft Access включите переключатель *Открыть базу данных*, выберите ранее созданную базу *Комплекты* и щелкните на кнопке *ОК*.

3. В окне *Комплекты*: база данных откройте панель *Запросы*. Дважды щелкните на значке *Создание запроса* в режиме *Конструктора* - откроется бланк запроса по образцу.

4. Согласно упражнению 3, создайте запрос на выборку, основанный на таблице *Процессоры*, в который войдут следующие поля:

- *Компонент*;
- *Модель*;
- *Цена отпоявая*;
- *Поставщик*;
- *Телефон*.

5. В строке *Условие отбора* поля *Компонент* введите *Процессор*.

6. Строку *Условие отбора* для поля *Цена отпоявая* надо заполнить таким образом, чтобы при запуске запроса пользователь получал предложение ввести нужное значение.

Текст, обращенный к пользователю, должен быть заключен в квадратные скобки. Если бы мы хотели отобразить процессоры, цена которых больше 100 условных единиц, мы бы написали >100. Если бы нам были нужны процессоры дешевле 80 единиц, мы бы написали < 80. Но если мы хотим дать пользователю возможность выбора, мы должны написать < [Введите максимальную цену].

7. Закройте запрос. При закрытии сохраните его под именем *Выбор комплектующих*.

8. В окне *Комплекты*: база данных откройте панель *Запросы* и запустите запрос *Выбор комплектующих* - на экране появится диалоговое окно *Введите значение параметра*.

9. Введите какое-либо число и щелкните на кнопке *ОК*. В зависимости от того, что реально содержится в таблице *Поставщики*, по результатам запроса будет сформирована результирующая таблица.

10. Закройте все объекты базы данных. Закройте программу Microsoft Access.

**Вывод.** Мы научились формировать запросы "с параметром" и узнали, что в основе этого вида запросов лежат запросы на выборку, у которых в поле *Условие отбора* записан заключенный в квадратные скобки текст, обращенный к пользователю.

##### **СОЗДАНИЕ ИТОГОВОГО ЗАПРОСА**

Если полностью заполнить данными таблицу *Комплекты*, введя параметры всех компонентов, входящих в сборочную спецификацию персонального компьютера, то можно узнать, во что обходится самостоятельность комплектующих узлов. Запросы, выполняющие вычисления по всем записям для какого-либо числового поля, называются *итоговыми запросами*. В итоговом запросе может рассчитываться сумма значений или величина среднего значения по всем ячейкам поля, может выбираться максимальное или минимальное значение данных в поле или выполняться иная *итоговая функция*. Итоговые запросы, как и запросы на выборку, готовятся с помощью бланка запроса по образцу.

Предположим, что малое предприятие собирает компьютеры трех классов: "Элитный", "Деловой" и "Экономичный". Несмотря на то, что архитектура всех компьютеров близка, их компоненты заметно отличаются по цене и техническим параметрам. Соответственно, имеются значительные отличия в цене этих трех моделей, что важно для захвата разных секторов рынка. Наша задача - подготовить итоговый отчет, с помощью которого можно определять цену каждой из модели компьютеров и динамично ее отслеживать при изменении входящих компонентов или их поставщиков.

1. Запустите программу Microsoft Access-2010 (Пуск → Программы → Microsoft Access).
2. В окне Microsoft Access включите переключатель *Открыть базу данных*, выберите ранее созданную базу *Комплектующие: база данных* и щелкните на кнопке *OK*.
3. В окне *Комплектующие: база данных* откройте панель *Таблицы*. Выберите таблицу *Комплектующие*.

4. Щелчком на значке *Конструктор* откройте таблицу в режиме проектирования - нам это необходимо для создания дополнительного поля *Класс*, в котором будут храниться данные о том, для какого класса изделий предназначены компоненты.

5. В начало структуры таблицы вставьте новое поле. Для этого выделите первое поле *Компонент* и нажмите клавишу *INSERT*.

6. Введите имя нового поля - *Класс* - и определите его тип - *Текстовый*.

7. Закройте окно *Конструктора*. При закрытии подтвердите необходимость изменить структуру таблицы.

8. Откройте таблицу *Комплектующие* и наполните ее содержанием, введя для каждого класса данные по следующим изделиям:

- материнская плата;
- процессор;
- оперативная память;
- жесткий диск;
- корпус;
- дисконд CD-ROM;
- дисконд гибких дисков;
- видеодантер;
- звуковая карта;
- клавиатура;
- мышь.

Цены на эти изделия для каждого класса проставьте произвольно. Прочие поля таблицы можно не заполнять - в формировании итогового запроса они участвовать не будут.

9. Закройте таблицу *Комплектующие*.

10. Откройте панель *Запросы* щелчком на одноименной кнопке окна *Комплектующие: база данных*.

11. Выполните двойной щелчок на значке *Создание запроса в режиме конструктора*. В открывшемся диалоговом окне *Добавление таблицы* выберите таблицу *Комплектующие*, на основе которой будет разрабатываться итоговый запрос. Закройте окно *Добавление таблицы*.

12. В бланк запроса по образцу введите следующие поля таблицы *Комплектующие: Класс, Компонент, Цена*.

13. Для поля *Класс* включите сортировку по возрастанию. Включите также сортировку по полю *Цена*, но на этот раз - по убыванию.

14. На панели инструментов Microsoft Access щелкните на кнопке *Групповые операции* или воспользуйтесь строкой меню (*Вид* → *Групповые операции*). Эта команда необходима для создания в нижней части бланка строки *Групповые операции*. Именно на ее базе и создаются итоговые вычисления. Все поля, отобранные для запроса, получают в этой строке значение *Группировка*.

15. Для поля, по которому производится группировка записей (в нашем случае - *Класс*), оставьте в строке *Групповые операции* значение *Группировка*. Для остальных полей щелкните на этой строке - появится кнопка раскрывающегося списка, из которого можно выбрать итоговую функцию для расчета значений в данном поле.

16. Для поля *Цена* выберите итоговую функцию *Sum* для определения стоимости изделия как суммы стоимостей комплектующих.

17. Для поля *Компонент* выберите итоговую функцию *Count*, определяющую общее количество записей, вошедших в группу. В нашем случае это количество узлов, из которых собран компьютер.

18. Закройте бланк запроса по образцу и дайте ему имя *Расчет стоимости изделия*. Запустите запрос и убедитесь, что он правильно работает.

19. Закройте все объекты базы данных. Завершите работу с программой Microsoft Access.

#### **Контрольные вопросы:**

1. Для чего необходима связь в таблицах?
2. Как создать запрос на выборку данных?
3. Как создать запрос с «параметром»?
4. Как создать итоговый запрос?

## Практическая работа №3 «Приведение БД к нормальной форме 3НФ».

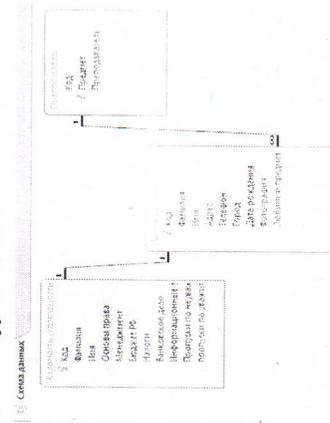
### Тема 1.1 Основы хранения и обработки данных. Проектирование БД.

Цель работы: «Проектирование реляционной БД»  
Материально-техническое обеспечение: Компьютер, операционная система Windows 7, MS Access 2010

#### Краткие теоретические сведения:

##### Связи между таблицами

Для полноценной работы Базы данных нужно наладить связи между таблицами, сделать их зависимыми друг от друга, чтобы по данным одной таблицы можно было узнать данные другой таблицы.



Связь между таблицами осуществляется с помощью числового идентификатора. Взяв идентификатор из одной таблицы, можно найти данные в другой таблице. Для установкой связи между двумя таблицами необходимо в группе **Показать или скрыть**, расположенной на вкладке **Работа с базами данных**, выбрать команду **Схема данных**. Появится окно **Схема данных**, в котором отображаются зависимости между таблицами.

Добавить новую связь можно, дважды щелкнув левой кнопкой мыши по окну или выбрав соответствующую команду на вкладке **Работа с базами данных** → **Схема данных** → **Изменить связь**. Появится окно **Изменение связей**, в котором нужно указать таблицы между которыми устанавливаются связи.

Как же влияет установление связи в таблицах на работу Microsoft Access?

Программ контролирует ввод данных в связанные таблицы по следующим правилам:

- запрещено добавлять значение, отсутствующее в исходной таблице, в связанную таблицу. То есть, запрещен ввод нового любимого предмета в таблицу **Личные данные**, если в таблице **Преподаватели** этот предмет отсутствует;
- запрещено удаление в исходной таблице записей, на которые есть ссылка в связанной таблице;
- при изменении записей в исходной таблице меняются записи в связанной таблице.

Связанные записи можно удалять только через исходную таблицу.

#### Порядок выполнения практической работы:

1. Изучить предлагаемые задания.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

#### Задания для выполнения практической работы:

##### СОЗДАНИЕ ЗАПРОСА НА ВЫБОРКУ

В этом упражнении мы создадим запрос на выборку жестких дисков, имеющих емкость не менее 8 Гбайт при цене менее 150 условных единиц. Результирующая таблица должна содержать также адрес поставщика и номер его телефона.

1. Запустите программу Microsoft Access-2010 (Пуск → Программы → Microsoft Access).
2. В окне Microsoft Access включите переключатель *Открыть базу данных*, выберите ранее созданную базу *Комплектующие* и щелкните на кнопке *ОК*.
3. В окне *Комплектующие: база данных* откройте панель *Запросы*. Дважды щелкните на значке *Создание запроса* в режиме *Конструктора* - откроется бланк запроса по образцу. Одновременно с ним откроется диалоговое окно *Добавление таблицы*.
4. В окне *Добавление таблицы* выберите таблицу *Поставщики* и щелкните на кнопке *Добавить*. Закройте окно *Добавление таблицы*.
5. В списке полей таблицы *Поставщики* выберите поля, включаемые в результирующую таблицу: *Компонент*, *Модель*, *Цена отпоява*, *Поставщик*, *Телефон*. Выбор производите двойными щелчками на именах полей.
6. Задайте условие отбора для поля *Компонент*, в соответствующую строку введите *Жесткий диск*. Из таблицы будут выбираться не все изделия, а только жесткие диски.
7. Задайте условие отбора для поля *Цена отпоява*. В соответствующую строку введите < 150. Из таблицы будут выбираться только изделия, имеющие цену менее 150 условных единиц.
8. Нам еще надо задать условие отбора по основному потребительскому параметру - емкости жесткого диска. Однако в таблице *Поставщики* такого поля нет. С другой стороны, в ней есть поле *Модель*, которое однозначно определяет параметры изделия. Благодаря тому, что по полю *Модель* у нас установлена связь с таблицей *Комплектующие*, мы получаем возможность ввести в запрос поле *Основной параметр*, взяв его из другой таблицы.
9. Добавьте список полей таблицы *Комплектующие* в верхнюю часть бланка запроса по образцу. Для этого щелкните правой кнопкой мыши в верхней области бланка и в открывшемся контекстном меню выберите пункт *Добавить таблицу* - откроется уже знакомое нам окно *Добавление таблицы*. Выберите в нем таблицу *Комплектующие*.
10. Двойным щелчком на поле *Основной параметр* в списке полей таблицы *Комплектующие* введите это поле в бланк запроса по образцу.
11. В строке *Условие отбора* столбца *Основной параметр* введите условие >8 (емкость диска - более 8 Гбайт).
12. Закройте бланк запроса по образцу. При закрытии запроса введите его имя - *Выбор комплектующих*.
13. В окне *Комплектующие: база данных* откройте только что созданный запрос и введите в таблицу *Комплектующие* и *Поставщики* при их наполнении в упражнении 1. Если ни одно изделие не соответствует условию отбора и получившаяся результирующая таблица не имеет данных, откройте базовые таблицы и наполните их модельными данными, позволяющими проверить работу запроса.
14. По окончании исследований закройте все открытые объекты и завершите работу с программой Microsoft Access.

##### СОЗДАНИЕ ЗАПРОСОВ "С ПАРАМЕТРОМ"

Выше мы рассмотрели, как действует условие отбора, но должны отметить его существенный недостаток. Пользователь базы данных работает с запросами, которые ему подготовил разработчик. Если, например, разработчик предусмотрел запрос, отбирающий изделия, имеющие цену менее 150 условных единиц, то пользователь базы уже не в

состоянии отобразить изделия, цена которых менее 140 условных единиц, поскольку у него нет соответствующего запроса.

Специальный тип запросов, называемый запросами "с параметром", позволяет пользователю самому ввести критерий отбора данных на этапе запуска запроса. Этот прием обеспечивает гибкость работы с базой.

Создадим простой запрос, позволяющий отбирать процессоры, предельную цену которых пользователь может задать сам при запуске запроса.

1. Запустите программу Microsoft Access - 2010 (Пуск → Программы → Microsoft Access).

2. В окне Microsoft Access включите переключатель *Открыть базу данных*, выберите ранее созданную базу *Комплектующие* и щелкните на кнопке *OK*.

3. В окне *Комплектующие: база данных* откройте панель *Запросы*. Дважды щелкните на значке *Создание запроса* в режиме *Конструктора* - откроется бланк запроса по образцу.

4. Согласно упражнению 3, создайте запрос на выборку, основанный на таблице *Поставщики*, в который войдут следующие поля:

- *Компонент*;
- *Модель*;
- *Цена отпавая*;
- *Поставщик*;
- *Телефон*.

7. В строке *Условие отбора* поля *Компонент* введите *Процессор*.

8. Строку *Условие отбора* для поля *Цена отпавая* надо заполнить таким образом, чтобы при запуске запроса пользователь получал предложение ввести нужное значение.

Текст, обращенный к пользователю, должен быть заключен в квадратные скобки. Если бы мы хотели отобразить процессоры, цена которых больше 100 условных единиц, мы бы написали >100. Если бы нам были нужны процессоры дешевле 80 единиц, мы бы написали < 80. Но если мы хотим дать пользователю возможность выбора, мы должны написать < [Введите максимальную цену].

11. Закройте запрос. При закрытии сохраните его под именем *Выбор комплектующих*.

12. В окне *Комплектующие: база данных* откройте панель *Запросы* и запустите запрос *Выбор комплектующих* - на экране появится диалоговое окно *Введите значение параметра*.

13. Введите какое-либо число и щелкните на кнопке *OK*. В зависимости от того, что реально содержится в таблице *Поставщики*, по результатам запроса будет сформирована результирующая таблица.

14. Закройте все объекты базы данных. Закройте программу Microsoft Access.

**Вывод.** Мы научились формировать запросы "с параметром" и узнали, что в основе этого вида запросов лежат запросы на выборку, у которых в поле *Условие отбора* записан заключенный в квадратные скобки текст, обращенный к пользователю.

#### СОЗДАНИЕ ИТОГОВОГО ЗАПРОСА

Если полностью заполнить данными таблицу *Комплектующие*, введя параметры всех компонентов, входящих в сборочную спецификацию персонального компьютера, то можно узнать, во что обходится себестоимость комплектующих узлов. Запросы, выполняющие вычисления по всем записям для какого-либо числового поля, называются *итоговыми запросами*. В итоговом запросе может рассчитываться сумма значений или величина среднего значения по всем ячейкам поля, может выбираться максимальное или минимальное значение данных в поле или выполняться иная *итоговая функция*. Итоговые запросы, как и запросы на выборку, готовятся с помощью бланка запроса по образцу.

Предположим, что малое предприятие собирает компьютеры трех классов: "Элитный", "Деловой" и "Экономичный". Несмотря на то, что архитектура всех компьютеров близка,

их компоненты заметно отличаются по цене и техническим параметрам. Соответственно, имеются значительные отличия в цене этих трех моделей, что важно для захвата разных секторов рынка. Наша задача - подготовить итоговый отчет, с помощью которого можно определять цену каждой из модели компьютеров и динамично ее отслеживать при изменении входящих компонентов или их поставщиков.

7. Запустите программу Microsoft Access-2010 (Пуск → Программы → Microsoft Access). 8. В окне Microsoft Access включите переключатель *Открыть базу данных*, выберите ранее созданную базу *Комплектующие* и щелкните на кнопке *OK*.

9. В окне *Комплектующие: база данных* откройте панель *Таблицы*. Выберите таблицу *Комплектующие*.

10. Щелчком на значке *Конструктор* откройте таблицу в режиме проектирования - нам необходимо для создания дополнительного поля *Класс*, в котором будут храниться данные о том, для какого класса изделий предназначены компоненты.

11. В начало структуры таблицы вставьте новое поле. Для этого выделите первое поле *Компонент* и нажмите клавишу *INSERT*.

12. Введите имя нового поля - *Класс* - и определите его тип - *Текстовый*.

9. Закройте окно *Конструктора*. При закрытии подтвердите необходимость изменить структуру таблицы.

10. Откройте таблицу *Комплектующие* и наполните ее содержанием, введя для каждого класса данные по следующим изделиям:

- материнская плата;
- процессор;
- оперативная память;
- жесткий диск;
- корпус;
- дисконд CD-ROM;
- дисконд гибких дисков;
- видеоадаптер;
- звуковая карта;
- клавиатура;
- мышь.

Цены на эти изделия для каждого класса проставьте произвольно. Прочие поля таблицы можно не заполнять - в формировании итогового запроса они участвовать не будут.

16. Закройте таблицу *Комплектующие*.

17. Откройте панель *Запросы* щелчком на одноименной кнопке окна *Комплектующие: база данных*.

18. Выполните двойной щелчок на значке *Создание запроса в режиме конструктора*. В открывшемся диалоговом окне *Добавление таблицы* выберите таблицу *Комплектующие*, на основе которой будет разрабатываться итоговый запрос. Закройте окно *Добавление таблицы*.

19. В бланк запроса по образцу введите следующие поля таблицы *Комплектующие: Класс, Компонент, Цена*.

20. Для поля *Класс* включите сортировку по возрастанию. Включите также сортировку по полю *Цена*, но на этот раз - по убыванию.

21. На панели инструментов Microsoft Access щелкните на кнопке *Групповые операции* или воспользуйтесь строкой меню (*Вид* → *Групповые операции*). Эта команда необходима для создания в нижней части бланка строки *Групповые операции*. Именно на ее базе и создаются итоговые вычисления. Все поля, отобранные для запроса, получают в этой строке значение *Группировка*.

22. Для поля, по которому производится группировка записей (в нашем случае - *Класс*), оставьте в строке *Групповые операции* значение *Группировка*. Для остальных полей

щелкните на этой строке - появится кнопка раскрывающегося списка, из которого можно выбрать итоговую функцию для расчета значений в данном поле.

20. Для поля *Цена* выберите итоговую функцию *Sum* для определения стоимости изделия как суммы стоимостей комплектующих.

21. Для поля *Комплект* выберите итоговую функцию *Count*, определяющую общее количество записей, вошедших в группу. В нашем случае это количество узлов, из которых собран компьютер.

22. Закройте бланк запроса по образцу и дайте ему имя *Расчет стоимости изделия*. Запустите запрос и убедитесь, что он правильно работает.

23. Закройте все объекты базы данных. Завершите работу с программой Microsoft Access.

#### **Контрольные вопросы:**

1. Для чего необходима связь в таблицах?
2. Как создать запрос на выборку данных?
3. Как создать запрос с «параметром»?
4. Как создать итоговый запрос?

## Практическая работа №4 «Создание базы данных в среде разработки»

### Тема 1.2 *Разработка и администрирование БД*

**Цель работы:** «Задание ключей, создание основных объектов БД»

**Материально-техническое обеспечение:** Компьютер, операционная система Windows 7, MS Access 2010

#### **Краткие теоретические сведения:**

**База данных состоит из следующих компонентов:**

**Таблицы** – основные объекты базы данных (БД). В них хранятся данные. Реляционная база данных может иметь много взаимосвязанных таблиц. Сведения по разным вопросам следует хранить в разных таблицах. В БД столбцы называются *полями*, а строки – *записями*. Для работы таблицу необходимо открыть. Перед окончанием работы ее следует закрыть, предварительно сохранив все изменения.

**Запрос** – это средство, с помощью которого извлекается из базы данных информация, отвечающая определенным критериям. Результаты запроса представляют не все записи из таблицы, а только те, которые удовлетворяют запросу.

**Формы** – Обеспечивают более наглядную работу с таблицами, с помощью форм в базу вводят новые данные или просматривают имеющиеся.

**Отчеты** – средство представления данных таблиц. Отчеты могут быть оформлены надлежащим образом и распечатаны в том виде, в котором требуется пользователю.

**Макросы** – набор из одной или более макрокоманд, выполняющих определенные операции (открытие форм, печать отчетов)

**Модули** – это программы, написанные на языке программирования Visual Basic.

#### **Таблицы**

С таблицами можно работать в двух режимах — **таблицы и конструктора**.

**В режиме таблицы** можно начать ввод данных сразу, структура таблицы при этом будет создаваться автоматически. Полям присваиваются имена с последовательными номерами («Поле1», «Поле2» и т. д.).

На вкладке **Создание** в группе **Таблицы** щелкните **Таблица**.

Приложение Access создаст базу данных с пустой таблицей с именем «Таблица1» и откроет эту таблицу в режиме таблицы. При этом будет выделена первая пустая ячейка столбца **Добавить поле**.

5. Чтобы добавить данные, начните их вводить в первую пустую ячейку или вставьте данные из другого

6. Для переименования столбца (поля) дважды щелкните заголовок столбца и введите новое имя. Рекомендуется присваивать полям значимые имена, чтобы при просмотре области **списка полей** было понятно, что содержится в каждом поле.

7. Для перемещения столбца сначала щелкните его заголовок, чтобы выделить столбец, а затем перетащите столбец в нужное место.

8. Кроме того, можно выбрать сразу несколько смежных столбцов, а затем одновременно перетащить их в новое место. Чтобы выбрать несколько последовательно расположенных столбцов, щелкните заголовок первого столбца, а затем, удерживая нажатой клавишу SHIFT, щелкните заголовок последнего столбца.

#### **Создание таблицы в режиме конструктора**

В режиме конструктора сначала создается структура новой таблицы. Затем можно переключиться в режим таблицы для ввода данных или ввести данные, используя другой метод, например, вставку или импорт.

4. На вкладке **Создание** в группе **Таблицы** щелкните **Конструктор таблиц**.

5. Для каждого поля в таблице введите имя в столбце **Имя поля**, а затем в списке **Тип данных** выберите тип данных.

6. Когда все необходимые поля будут добавлены, сохраните таблицу.

4 Щелкните значок **Кнопка Microsoft Office** , а затем выберите команду **Сохранить**.

**Порядок выполнения практической работы:**

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

**Задания для выполнения практической работы:**

**ЗАДАНИЕ 1. СОЗДАНИЕ СТРУКТУРЫ БД.**

**1.1.** Запустить программу Microsoft Access для создания новой базы данных. Для этого следует нажать кнопки сначала **Пуск**, затем **Программы**, среди которых нужно выбрать **Microsoft Access**.

**1.2.** Создать новую базу данных под названием **МОДЕЛИ**, а в скобках указать **Фамилию ученика**.

Для создания базы данных войдите в **Пуск/Программы/Microsoft Access**. После чего откроется диалоговое окно на экране монитора **Создание новой базы данных**. Нужно выбрать пункт **Новая база данных** и нажать **ОК**, в появившемся окне выбрать диск **C:/Мои документы** и в строке **Имя файла** следует ввести слово **Модели(Фамилия ученика)** и затем нажать кнопку **Создать**. После чего открывается окно базы данных под названием **Модели(Фамилия ученика)**. В верхней части окна базы данных представлено меню, где перечислены объекты базы данных:

- Таблицы,
- Запросы,
- Формы,
- Отчеты,
- Макросы,
- Модули.

**1.3.** Создать структуру новой таблицы.

Для создания структуры новой таблицы в окне базы данных следует щелкнуть мышью по кнопке **Таблицы**, затем по кнопке **Создать**. После чего **Система** откроет диалоговое окно, в котором будет предложено воспользоваться одним из следующих способов создания таблицы:

- Режим таблицы,
- Конструктор,
- Мастер таблиц,
- Импорт таблиц,
- Связь с таблицами.

Нужно выделить способ **Конструктор**, так как будем конструировать свою таблицу, и нажав **Ок**, структура **Таблицы** открывается в режиме конструктора, куда необходимо ввести

- Имя поля,
- Тип данных,
- Описание.

**1.4.** Ввести в новую таблицу имена полей и их спецификации так, как это показано в приведенной ниже таблице.

ИМЯ ПОЛЯ	ТИП ДАННЫХ	ОПИСАНИЕ
Код модели	Числовой	Код модели по заводскому заказу
Модель	Текстовый	Тип салона
Мощность	Текстовый	Мощность двигателя (квт/л.с.)
Количество дверей	Числовой	Двухдверный или четырехдверный Салон
Коробка передач	Текстовый	Автоматическая или ручная
Обивка	Текстовый	Велюр, кожа, ткань
Другое оснащение	Мето	Дополнительные аксессуары
Изображение	Поле объекта OLE	Рисунок, представляющий модель
Заводская цена	Денежный	Заводская продажная нетто-цена
Транспортные издержки	Денежный	Издержки на доставку
Препродажные издержки	Денежный	Издержки на продажную подготовку

1.5. Ввести свойства полей новой таблицы так, как это показано ниже (приведены только те свойства полей, значения которых отличаются от стандартных).

Поле	Размер Поля	Формат Поля	Число десятичных Знаков	Значение По умолчанию	Обязательное Поле	Индексированное поле
Код модели	Длинное целое	Основной	0		Да	Да(Не допускаются совпадения)
Модель	20				Да	Да(совпадения допуск)
Мощность	10					Нет
Количество Дверей	Целое	Основной	0	4		Нет
Коробка Передач	15			Ручная		Нет
Обивка	20					Нет
Другое Оснащение	Мето					
Изображение					Нет	
Заводская Цена		Денежный	Авто		Да	Нет
Транспортные Издержки		Денежный	Авто		Да	Нет
Препродажные Издержки		Денежный	Авто	105	Да	Нет

**1.6.** Установить первичный ключ таблицы.

Для этого необходимо при выделении поля **Код модели** нажать на панели инструментов **Ключевое поле**.

1.7. Сохранить созданную Структуру таблицы под именем Предлагаемые модели и закрыть таблицу.

1.8. Открыть снова таблицу Предлагаемые модели в режиме Конструктор и отредактировать ее:

- ввести новую строку под названием Цвет перед строкой Количество дверей.
- Для вставки нового поля перед существующим необходимо выбрать область панели инструментов. В новую строку ввести название имя поля –Цвет, тип данных- текстовый, описание – цвет кузова.

- изменить название поля **Мощность** на название **Мощность двигателя**.

- поменять местами поля **Цвет** и **Количество дверей**.

Для того, чтобы поменять местами предлагаемые поля, необходимо перед полем Цвет создать пустую строку, как это было указано выше, затем вырезать строку Количество дверей и вставить ее в созданную пустую строку, используя панель инструментов.

**Контрольные вопросы:**

1. Опишите этапы создания таблицы в конструкторе?
2. Какие типы данных в БД вы знаете?
3. Какие параметры форматирования таблиц вы можете применить в БД?

Практическая работа №5 «Организация локальной сети. Настройка локальной сети»

Тема 1.2 *Разработка и администрирование БД*

Цель работы: «создание проекта БД»

Материально-техническое обеспечение: Компьютер, операционная система Windows 7, VFoxPro

**Краткие теоретические сведения:**

Основные типы полей, используемые в Microsoft Visual FoxPro приведены в таблице

Тип поля	Содержимое
Character (Символьный)	Любые символы. Максимальный размер символьного поля – 254 символа
Numeric (Числовой)	Включает цифры от 0 до 9, знак (необязательно), дробную часть (необязательно)
Float (Вещественный)	Числа, имеющие дробную часть(представляются в формате с плавающей точкой), определенные с обычной точностью
Double (Двойной Вещественный сдвойной точностью)	Числа, имеющие дробную часть(представляются в формате с плавающей точкой), определенные с двойной точностью
Integer (Числовой, целочисленный)	Целые числа
Data (Дата)	Дата, определяющая день, месяц и год. Ввод в это поле контролируется VisualFoxPro автоматически
Data Time (Дата и время)	Дата и время. По умолчанию формат вводимого значения имеет вид мм/дд/гг гг.мм.сс
Logical (Логический)	Логические данные. Может принимать только Т. – истина, F. – ложь
Мемо (Текстовое поле произвольной длины)	Примечания, размер которых неограничен. Может содержать любую символьную информацию, размер которой ограничивается лишь объемом доступного дискового пространства. Фактически – это указатель на блок данных в файле с расширением frt (файлы dbf и frt одноименны). Для заполнения данного поля нужно дважды щелкнуть мышью по данному полю (или CTRL+Home), после чего в открывшемся окне набрать нужный текст примечаний. Закрыв окно редактирования (CTRL+W), значение поля будет сохранено, а слово memo в поле базы данных изменится на Memo.
General (Общие, двойное поле произвольной длины)	Поле, используется для хранения ссылки на объект. Под объектом понимается любой текстовый файл, звуковой файл, диаграмма, иллюстрация или другой файл, созданный с помощью приложения, поддерживающего технологию OLE. Для заполнения данного поля нужно двойным щелчком мыши на нем открыть окно редактирования. После чего выбрать пункт меню Edit(правка)-insert object(вставить объект) и в открытом выбранном приложении создать нужный объект

	(например, рисунок). Закрыв окно приложения и окно редактирования поля, поле gen будет заменено на Gen, значение будет сохранено
Currency (Денежный)	Поле для представления денежных сумм, используется для хранения чисел с четырьмя знаками после запятой
Memobinary	Двоичный текст произвольной длины. Содержит любые данные поля memo, которые можно использовать, не изменяя кодовую страницу
Character binary	Двоичный текст длиной до 254 символов. Содержит любые символьные данные, которые используются, без изменения кодовой страницы

**Порядок выполнения практической работы:**

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

**Задания для выполнения практической работы:**

Запустим Microsoft Visual FoxPro

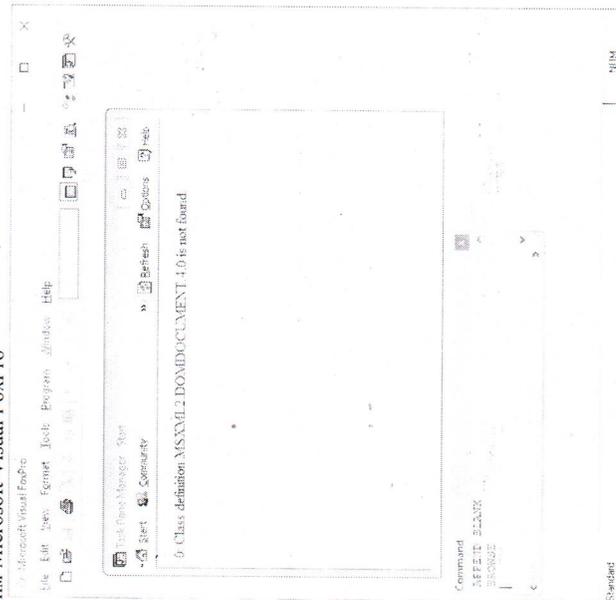


Рисунок 1

У нас появится 2 окна

1. Command – В этом окне мы будем вводить команды
2. Task Pane Manager – окно с инструкцией. Прежде чем начать работать в FoxPro надо ознакомиться с основными командами для создания "Базы данных", Create <имя> — создать

Modify structure – изменить структуру

Use < имя> — открыть подготовит

Select < имя> — выбрать

Browse – открыть визуализировать

Append blank – добавить запись

Начнём создания нашей первой Базы данных. Для этого напишем Create <имя> в окне команд. У нас откроется окно структуры базы данных

Table Designer - 777.dbf

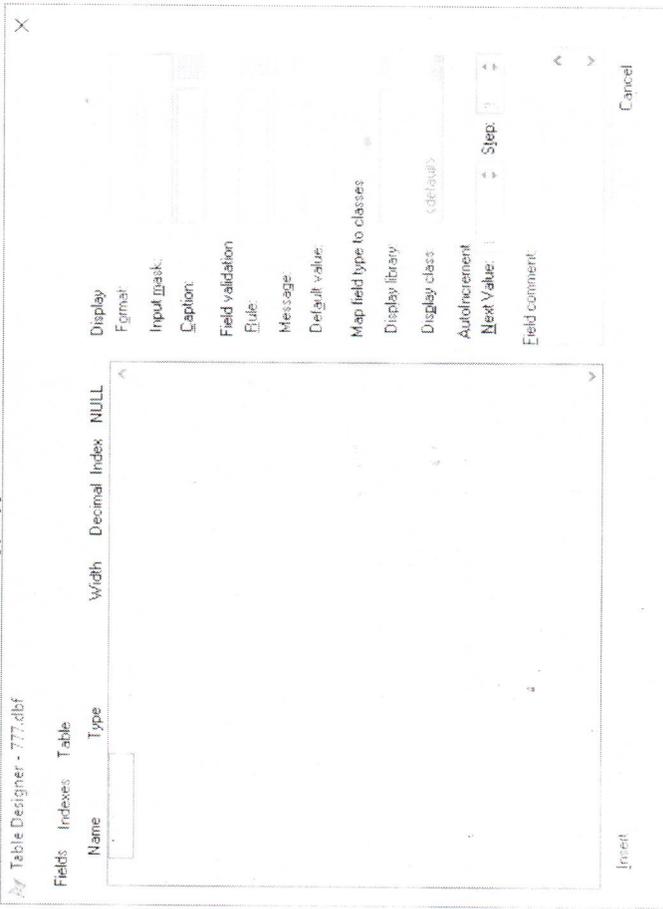


Рисунок 2

Name это название будущего столбца

Type это тип столбца

Width – ширина столбца

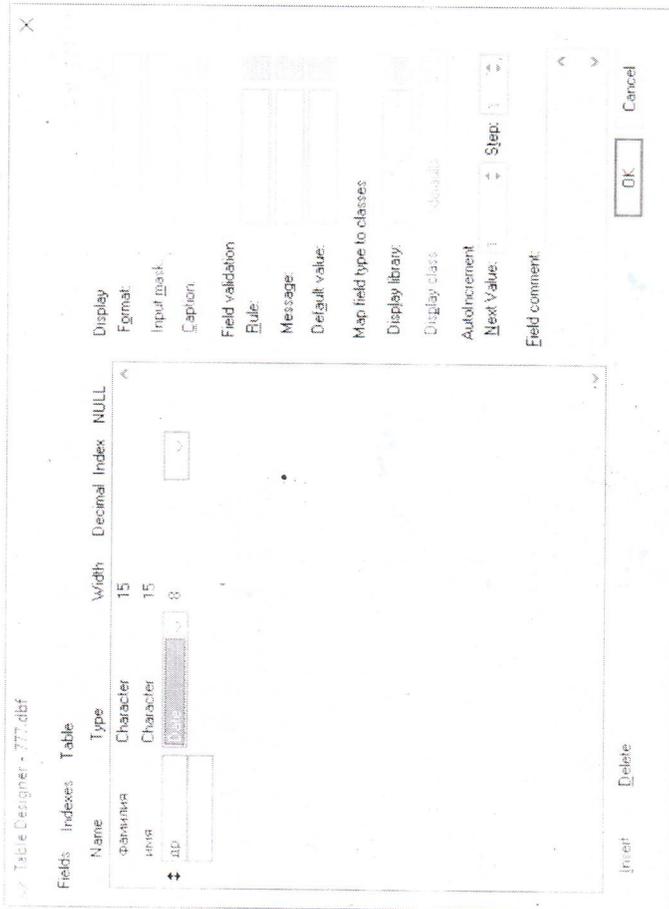


Рисунок 3

Основные типы полей, используемые в Microsoft Visual FoxPro приведены в таблице

Тип поля	Содержимое
Character (Символьный)	Любые символы. Максимальный размер символьного поля – 254 символа
Numeric (Числовой)	Включает цифры от 0 до 9, знак (необязательно), дробную часть (необязательно)
Float (Вещественный)	Числа, имеющие дробную часть (представляются в формате с плавающей точкой), определенные с обычной точностью
Double (Двойной Вещественный с двойной точностью)	Числа, имеющие дробную часть (представляются в формате с плавающей точкой), определенные с двойной точностью
Integer (Числовой, целочисленный)	Целые числа
Data (Дата)	Дата, определяющая день, месяц и год. Ввод в это поле контролируется VisualFoxPro автоматически
DateTime (Дата и время)	Дата и время. По умолчанию формат вводимого значения имеет вид мм/дд/гг чч.мм.сс

Logical (Логический)	Логические данные. Может принимать только Т. – истина, F. – ложь
Мемо (Текстовое поле произвольной длины)	Примечания, размер которых неограничен. Может содержать любую символьную информацию, размер которой ограничивается лишь объемом доступного дискового пространства. Фактически – это указатель на блок данных в файле с расширением frt (файлы dbf и frt одноименны). Для заполнения данного поля нужно дважды щелкнуть мышью по данному полю (или CTRL+Home), после чего в открывшемся окне набрать нужный текст примечаний. Закрыв окно редактирования (CTRL+W), значение поля будет сохранено, а слово мемо в поле базы данных изменится на Memo.
General (Общие, двоичное поле произвольной длины)	Поле, используется для хранения ссылки на объект. Под объектом понимается любой текстовый файл, звуковой файл, диаграмма, иллюстрация или другой файл, созданный с помощью приложения, поддерживающего технологию OLE. Для заполнения данного поля нужно двойным щелчком мыши на нем открыть окно редактирования. После чего выбрать пункт меню Edit(правка)-insert object(вставить объект) и в открытом выбранном приложении создать нужный объект (например, рисунок). Закрыв окно приложения и окно редактирования поля, поле gen будет заменено на Gen, значение будет сохранено
Currency (Денежный)	Поле для представления денежных сумм, используется для хранения чисел с четырьмя знаками после запятой
Memobinary	Двоичный текст произвольной длины. Содержит любые данные поля мемо, которые можно использовать, не изменяя кодовую страницу
Character binary	Двоичный текст длиной до 254 символов. Содержит любые символьные данные, которые используются, без изменения кодовой страницы

Чтобы визуализировать нашу базу данных надо ввести Browse

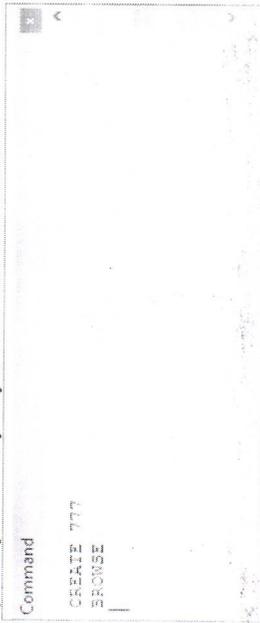


Рисунок 4

Далее у нас откроется наша таблица но редактировать мы там нечего не можем. Для этого пишем команду Append blankкаждый раз когда хотим добавить строчку и опять повторяем команду Browse

Рисунок 6

Всё готово теперь мы можем вносить данные в нашу базу данных. Важно – дата пишется мм/дд/гг

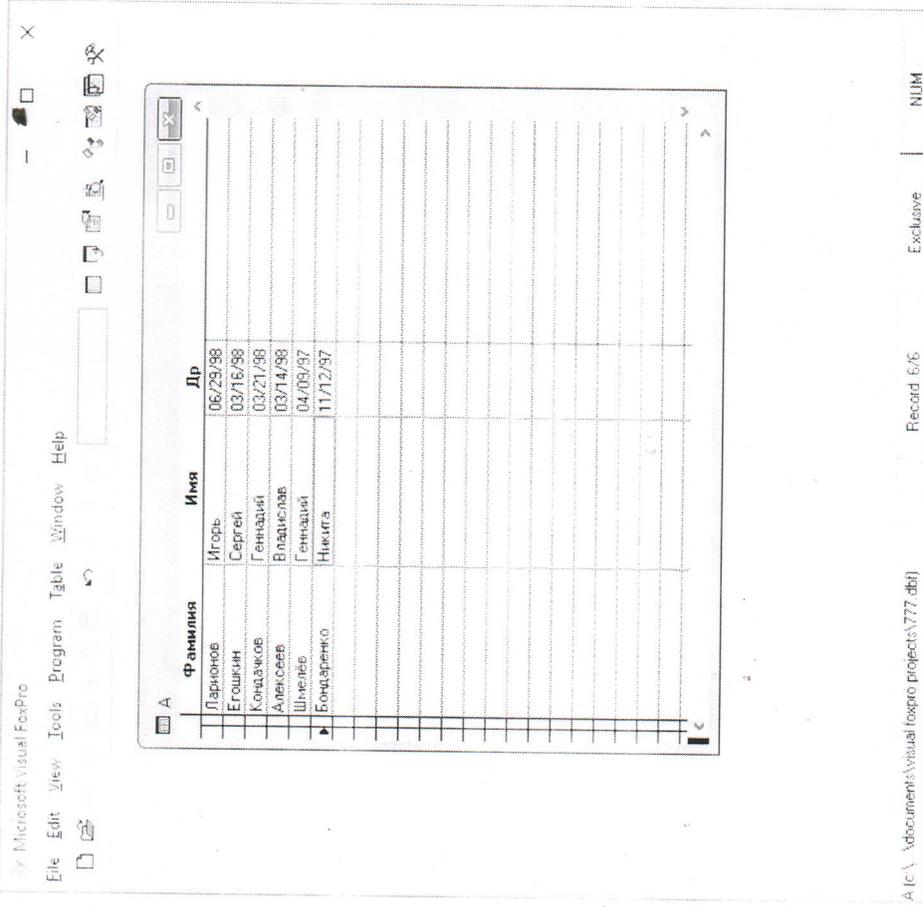


Рисунок 7

**Контрольные вопросы:**

1. Как создать таблицу в FoxPro?
2. Какие типы полей используются в FoxPro?

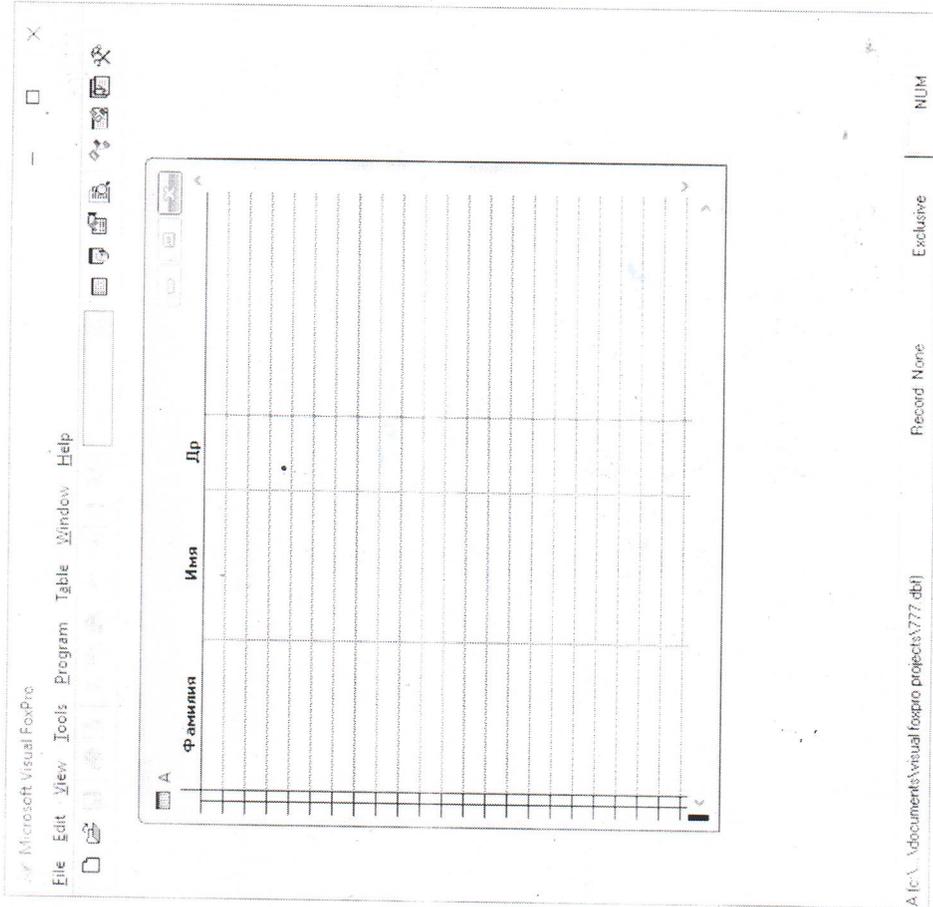


Рисунок 5







Name	Surname	Birthday	Telephone
Иван	Иванов	04/28/98	9521697603
Владислав	Алексеев	05/25/98	9512586969
Игорь	Париков	12/01/98	9514789566
Геннадий	Шмелев	06/14/98	9521697587
Никита	Бондаренко	04/15/98	9531587987
Владислав	Дубовик	04/01/98	9231234564
Жердеев	Евгений	07/03/98	9235498798
Королев	Алексей	01/01/98	9238598986
Давыдов	Никита	02/03/98	9234564567
Кондажов	Геннадий	02/04/98	9231231231
Климентьев	Дмитрий	04/25/98	9216598788
Ястребова	Анастасия	01/25/98	8945987989
Петр	Петр	/ /	/ /
Иван	Иванов	/ /	/ /

Рисунок 4 (Результат выполнения команды)

И так, тут мы видим, что 1 строка в нашей БД помечилась «Черным прямоугольником» Ее нам и надо удалить.  
В этом нам поможет команда «Pack-Удаление помеченных записей» (Рисунок 5)



Name	Surname	Birthday	Telephone
Владислав	Алексеев	05/25/98	9512586969
Игорь	Париков	12/01/98	9514789566
Геннадий	Шмелев	06/14/98	9521697587
Никита	Бондаренко	04/15/98	9531587987
Владислав	Дубовик	04/01/98	9231234564
Жердеев	Евгений	07/03/98	9235498798
Королев	Алексей	01/01/98	9238598986
Давыдов	Никита	02/03/98	9234564567
Кондажов	Геннадий	02/04/98	9231231231
Климентьев	Дмитрий	04/25/98	9216598788
Ястребова	Анастасия	01/25/98	8945987989

Рисунок 5 (Результат выполнения команды)

Как вы видите, что после выполнения команды Pack, строка пропала и после её выполнения, ваша БД закроется, для этого воспользуйтесь командой «Browse» она откроет вам вашу БД.

**Задание 4**

Добавим в нашу БД двух Ивановых (Петра и Ивана), с помощью команды REPLACE name WITH (Рисунок 6)



Name	Surname	Birthday	Telephone
Владислав	Алексеев	05/25/98	9512586969
Игорь	Париков	12/01/98	9514789566
Геннадий	Шмелев	06/14/98	9521697587
Никита	Бондаренко	04/15/98	9531587987
Владислав	Дубовик	04/01/98	9231234564
Жердеев	Евгений	07/03/98	9235498798
Королев	Алексей	01/01/98	9238598986
Давыдов	Никита	02/03/98	9234564567
Кондажов	Геннадий	02/04/98	9231231231
Климентьев	Дмитрий	04/25/98	9216598788
Ястребова	Анастасия	01/25/98	8945987989
Петр	Петр	/ /	/ /
Иван	Иванов	/ /	/ /

Рисунок 6 (Команда для добавления имени)

Теперь добавим фамилии, при помощи команды REPLACE surname WITH (Рисунок 7)



Name	Surname	Birthday	Telephone
Владислав	Алексеев	05/25/98	9512586969
Игорь	Париков	12/01/98	9514789566
Геннадий	Шмелев	06/14/98	9521697587
Никита	Бондаренко	04/15/98	9531587987
Владислав	Дубовик	04/01/98	9231234564
Жердеев	Евгений	07/03/98	9235498798
Королев	Алексей	01/01/98	9238598986
Давыдов	Никита	02/03/98	9234564567
Кондажов	Геннадий	02/04/98	9231231231
Климентьев	Дмитрий	04/25/98	9216598788
Ястребова	Анастасия	01/25/98	8945987989
Петр	Петр	/ /	/ /
Иван	Иванов	/ /	/ /

Рисунок 7 (Результат выполнения команды)

Также добавим номер телефона, командой REPLACE secondname WITH (Рисунок 8)

```

Command
REPLACE Telephone WITH 9521645602
REPLACE Telephone WITH 9521645602

```

Name	Surname	Birthday	Telephone
Владислав	Алексеев	05/25/98	9521645602
Игорь	Ларионов	12/01/98	9514789566
Геннадий	Шмелев	06/14/98	9521697587
Никита	Бондаренко	04/15/98	9531587987
Владислав	Дубовик	04/01/98	9231234564
Жерард	Евгений	07/03/98	9235498798
Королёв	Алексей	01/01/98	9236598986
Давид	Никита	02/03/98	9234564567
Константино	Геннадий	02/04/98	9231231231
Климентьев	Дмитрий	04/25/98	9216598788
Ястребова	Анастасия	01/25/98	8845987989
Петр	Иванов	/ /	9521697602
Иван	Иванов	/ /	9521645602

Рисунок 8 (Результат выполнения команды)

Добавляем дату рождения командой REPLACE birthday WITH (Рисунок 9)

```

Command
REPLACE birthday WITH (^1998-01-02)
REPLACE birthday WITH (^1998-05-02)

```

Name	Surname	Birthday	Telephone
Владислав	Алексеев	05/25/98	9521645602
Игорь	Ларионов	12/01/98	9514789566
Геннадий	Шмелев	06/14/98	9521697587
Никита	Бондаренко	04/15/98	9531587987
Владислав	Дубовик	04/01/98	9231234564
Жерард	Евгений	07/03/98	9235498798
Королёв	Алексей	01/01/98	9236598986
Давид	Никита	02/03/98	9234564567
Константино	Геннадий	02/04/98	9231231231
Климентьев	Дмитрий	04/25/98	9216598788
Ястребова	Анастасия	01/25/98	8845987989
Петр	Иванов	01/02/98	9521697602
Иван	Иванов	05/02/98	9521645602

Рисунок 9 (Результат выполнения команды)

Задание 5  
Добавляем поле доход и заполняем его (произвольно) (Рисунок 10)

Фамилия	Имя	Дп	Телефон	Доход
Алексеев	Владислав	05/25/98	9500319831	5000
Ларионов	Игорь	12/01/98	9500319831	1000
Шмелев	Геннадий	06/14/98	9500319831	1000
Бондаренко	Никита	04/15/98	9500319831	1100
Дубовик	Владислав	04/01/98	9500319831	1100
Евгений	Жерард	07/03/98	9500319831	2000
Алексей	Королёв	01/01/98	9500319831	1800
Никита	Давид	02/03/98	9500319831	1500
Геннадий	Константино	02/04/98	9500319831	1500
Дмитрий	Климентьев	04/25/98	9500319831	1500
Анастасия	Ястребова	01/25/98	9500319831	1500
Иванов	Петр	01/02/98	9500319831	1500
Иванов	Иван	05/02/98	9500319831	1500

REPLACE доход WITH 50000  
REPLACE доход WITH 10000  
REPLACE доход WITH 1100  
REPLACE доход WITH 1100  
REPLACE доход WITH 2000  
REPLACE доход WITH 1800  
REPLACE доход WITH 1500  
REPLACE доход WITH 1500



Рисунок 10 (Результат выполнения команды)

Задание 6  
Добавляем поле налог, поле налог 12% от дохода (Рисунок 11)

Фамилия	Имя	Дп	Телефон	Доход	Налог
Алексеев	Владислав	05/25/98	9500319831	5000	600
Ларионов	Игорь	12/01/98	9500319831	1000	120
Шмелев	Геннадий	06/14/98	9500319831	1000	120
Бондаренко	Никита	04/15/98	9500319831	1100	132
Дубовик	Владислав	04/01/98	9500319831	1100	132
Евгений	Жерард	07/03/98	9500319831	2000	240
Алексей	Королёв	01/01/98	9500319831	1800	216
Никита	Давид	02/03/98	9500319831	1500	180
Геннадий	Константино	02/04/98	9500319831	1500	180
Дмитрий	Климентьев	04/25/98	9500319831	1500	180
Анастасия	Ястребова	01/25/98	9500319831	1500	180
Иванов	Петр	01/02/98	9500319831	1500	180
Иванов	Иван	05/02/98	9500319831	1500	180

REPLACE налог WITH 0.12\*5000  
REPLACE налог WITH 0.12\*1000  
REPLACE налог WITH 0.12\*1000  
REPLACE налог WITH 0.12\*1100  
REPLACE налог WITH 0.12\*1100  
REPLACE налог WITH 0.12\*2000  
REPLACE налог WITH 0.12\*1800  
REPLACE налог WITH 0.12\*1500  
REPLACE налог WITH 0.12\*1500  
REPLACE налог WITH 0.12\*1500  
REPLACE налог WITH 0.12\*1500  
REPLACE налог WITH 0.12\*1500



**Контрольные вопросы:**

1. К какому виду СУБД относится FoxPro?
2. С какими командами вы научились сегодня работать?
3. Как добавить новое поле в БД?

Практическая работа №7 «Экспорт данных базы в документы пользователя».

**Тема 1.2 Разработка и администрирование БД**

**Цель работы:** «Преобразование реляционной БД»

**Материально-техническое обеспечение:** Компьютер, операционная система Windows 7, FoxPro

**Краткие теоретические сведения:**

**Команды для осуществления сортировки:**

Sort to <имя новой БД> on <имя поля> — сортирует и создаёт новую базу данных с уже отсортированными данными

Ключи команды: [A] [D] [C]

[A] — задает возрастающий порядок для данного поля

[D] — убывающий порядок

SET DATE — задает формат отображаемых выражений типа даты и DateTime.

**Порядок выполнения практической работы:**

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

**Задания для выполнения практической работы:**

**Сортировка базы данных**

**Задание:**

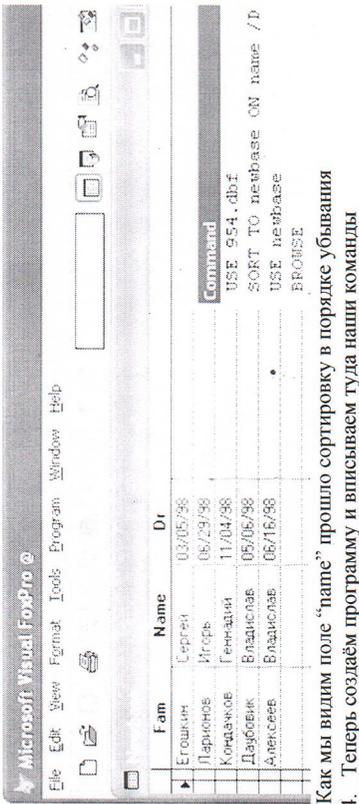
1. Отсортировать базу данных
1. Вызываем базу которую мы будем сортировать

```
Command  
USE 954.dbf
```

2. Сортируем базу 954.dbf по параметру "name" на убывание при этом создаём новую базу под названием "newbase"

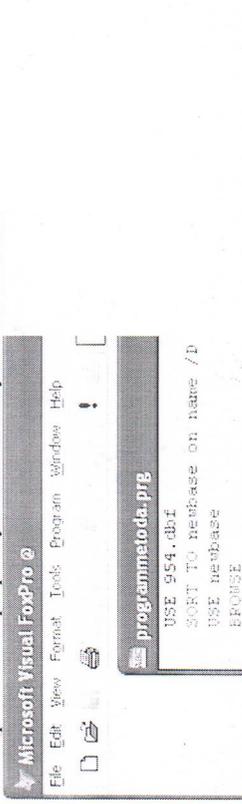
```
Command  
USE 954.dbf  
SORT TO newbase ON name /D
```

3. Вызываем базу созданную нами и проверяем сортировку

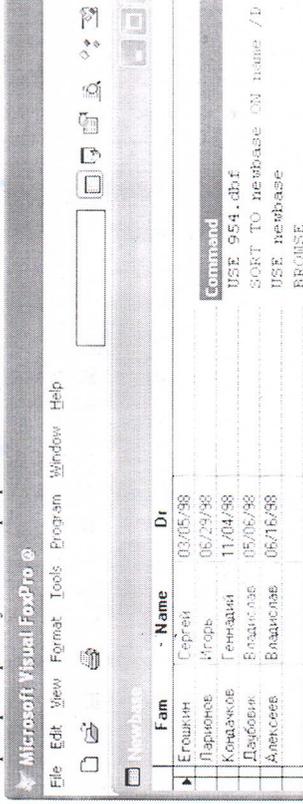


Как мы видим поле "name" прошло сортировку в порядке убывания

4. Теперь создаём программу и вписываем туда наши команды



5. Проверяем работу нашей программы



Чтобы проверить работоспособность программы нужно открыть пункт меню Program в нём выбрать Do... далее выбираем нашу программу и пишем ACTIVATE MENU <Название меню>

**Контрольные вопросы:**

1. Для чего предназначена сортировка?
2. Как осуществляется сортировка в FoxPro?
3. Как осуществить поиск в БД?

Практическая работа №8 «Импорт данных пользователя в базу данных».

### Тема 1.2 Разработка и администрирование БД

**Цель работы:** «Получить некоторые практические навыки по созданию меню в программе Visual FoxPro»

**Материально-техническое обеспечение:** Компьютер, операционная система Windows 7, FoxPro

**Краткие теоретические сведения:**

**Команды для осуществления сортировки:**

Sort to <имя новой БД> on <имя поля> — сортирует и создаёт новую базу данных с уже отсортированными данными

Ключи команды: [/A | /D] [/C]

[/A] — задает возрастающий порядок для данного поля

[/D] — убывающий порядок

SET DATE — задает формат отображаемых выражений типа даты и DateTime.

**Порядок выполнения практической работы:**

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

**Задания для выполнения практической работы:**

**Сортировка базы данных**

**Задание:**

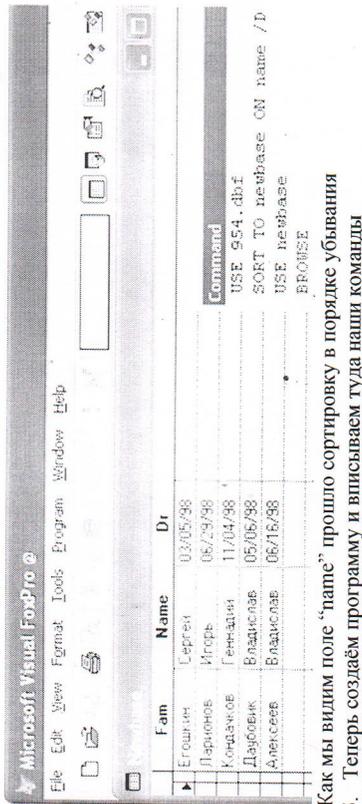
2. Отсортировать базу данных
2. Вызываем базу которую мы будем сортировать

```
Command
USE 954.dbf
```

3. Сортируем базу 954.dbf по параметру "name" на убывание при этом создаём новую базу под названием "newbase"

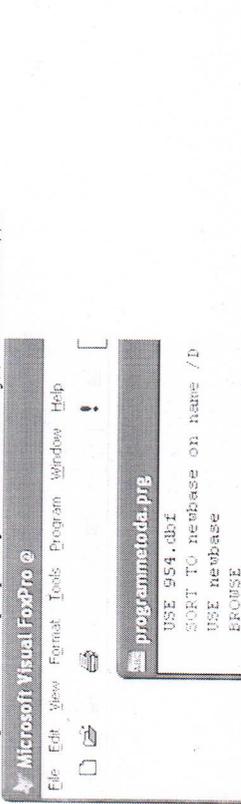
```
Command
USE 954.dbf
SORT TO newbase ON name /D
BROWSE
```

4. Вызываем базу созданную нами и проверяем сортировку

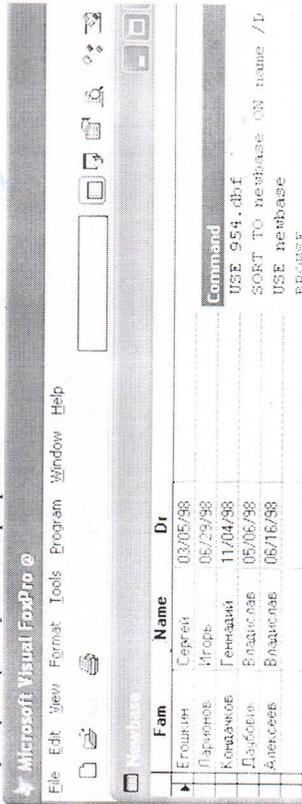


Как мы видим поле "name" прошло сортировку в порядке убывания

5. Теперь создаём программу и вписываем туда наши команды



6. Проверяем работу нашей программы



Чтобы проверить работоспособность программы нужно открыть пункт меню Program в нём выбрать Do... далее выбираем нашу программу и пишем ACTIVATE MENU <Название меню>

Создание поиска

Запускаем программу меню:

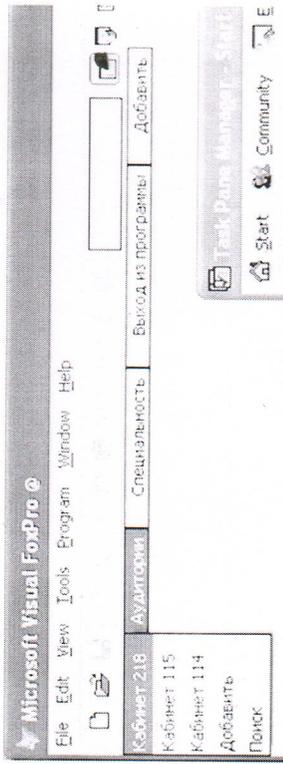


Рис.1(Наше меню.)

Далее нажимаем на кнопку «Поиск» дабы нам необходимо запустить нашу программу:

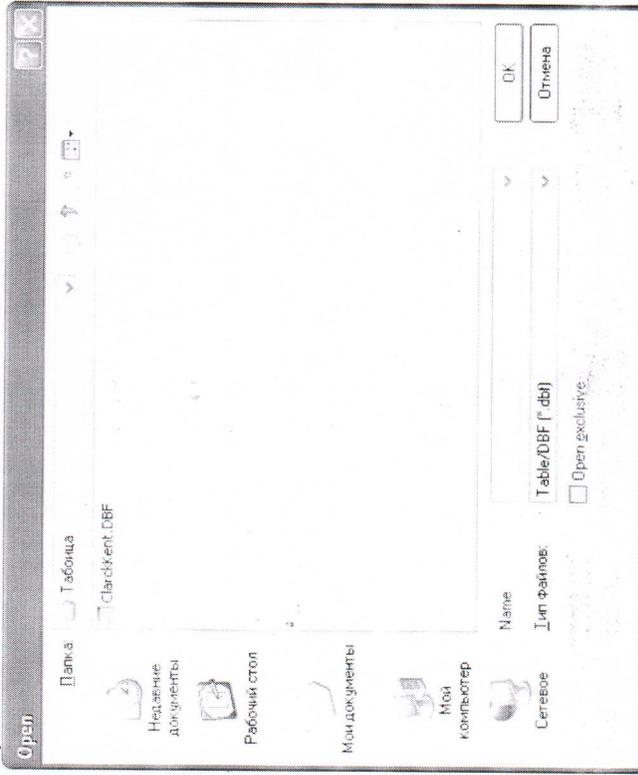


Рис.2(Открывается поиск нашей таблицы.)

Выбираем.

Рис.4(Ну и соответственно видим нашу выполненную работу.)  
И немного расскажем, как мы это сделали.

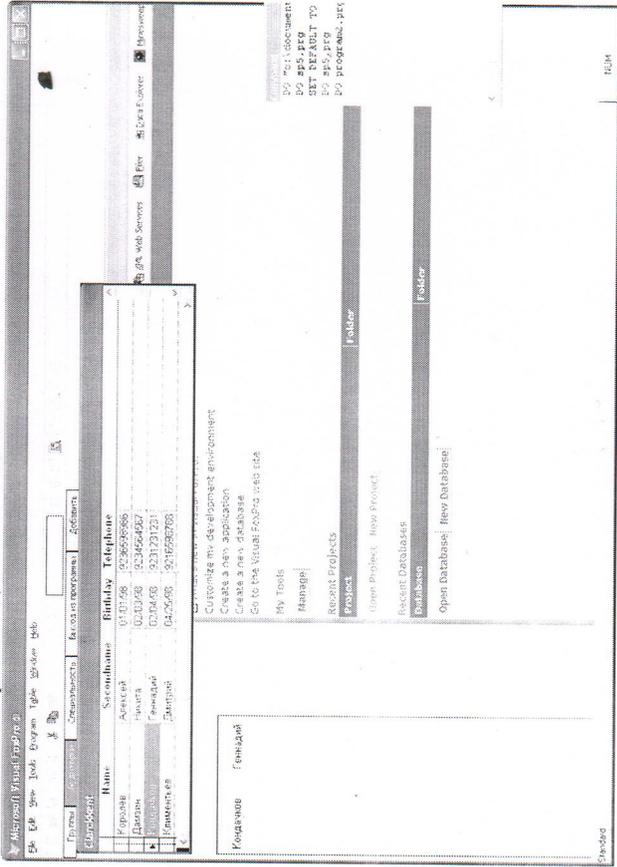


Рис.3(Открылась таблица.)  
Заходим в пункт меню «Поиск»

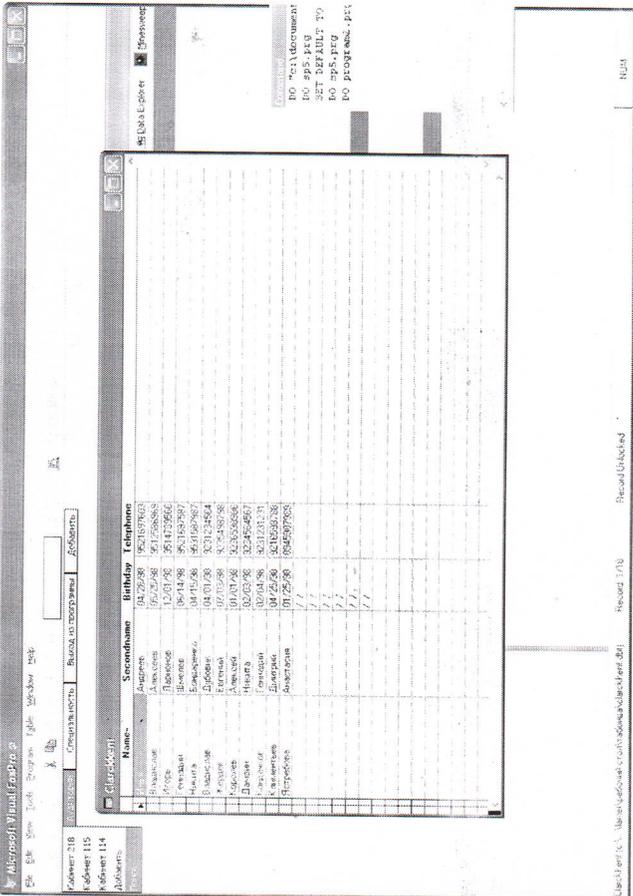


Рис.5(Добавление пункта)  
Пропишем в нашем меню пункт «Поиск» чтобы он обращался к нашей программе, которая будет выполнять поиск.

```

Ну, и конечно же, наша сама программа выполняющая поиск:
define prppr in
define bar 1 of v prppr "Кабинет 218"
define bar 2 of v prppr "Кабинет 115"
define bar 3 of v prppr "Кабинет 114"
define bar 4 of v prppr "Добавить"
define bar 5 of v prppr "Поиск"
on selection pad f2 of Andreev activate prppr v
on selection bar 1 of v do sp1
on selection bar 2 of v do sp2
on selection bar 3 of v do sp4
on selection bar 4 of v do sp4
on selection bar 5 of v do sp5
--

```

Рис.6(Программа)

**Контрольные вопросы:**

1. Для чего предназначена сортировка?
2. Как осуществляется сортировка в FoxPro?
3. Как осуществить поиск в БД?

Практическая работа №9 «Выполнение настроек для автоматизации обслуживания базы данных».

**Тема 1.2. Разработка и администрирование БД**

**Цель работы:** «Преобразование реляционной БД»

**Материально-техническое обеспечение:** Компьютер, операционная система Windows 7,

FoxPro

**Краткие теоретические сведения:**

**Порядок выполнения практической работы:**

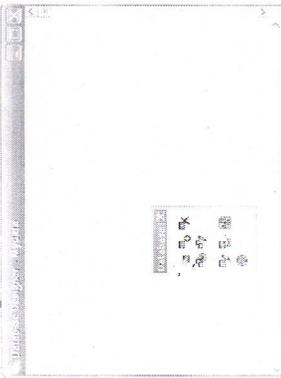
1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

**Задания для выполнения практической работы:**

«Создание базы данных и заполнение таблиц»

**1. Разработка структуры базы данных.**

Для создания новой БД выберите пункт меню File/New, в окне New установите флажок Database и нажмите кнопку New File. Для сохранения БД укажите путь к своей папке и введите имя новой БД - Mудata. Откроется окно Дизайнера БД.



Для работы с базой данных используется панель инструментов Дизайнера БД.

**Назначение кнопок:**

- создать новую таблицу
- добавить существующую таблицу
- удалить таблицу из БД
- создать новое удаленное представление
- создать новое локальное представление
- изменить структуру и свойства таблицы
- просмотр таблицы
- редактирование хранимых процедур
- создание удаленной БД

**2. Разработка структур таблиц базы данных.**

Создайте в БД первую таблицу (Типы книг) с именем tip.

Поля таблицы tip

Имя поля	Тип поля, размер	Описание (св-во Sortion)
Kodt	Integer (AutoInc)	Код типа
Name	Character, 30	Наименование типа

Создайте вторую таблицу (Книги) с именем knig.

Поля таблицы knig

Имя поля	Тип поля, размер	Описание (св-во Sortion)
Kodt	integer	Код типа
Kodkn	Integer (AutoInc)	Код книги
Autot	Character, 30	Автор
Nazvanie	Character, 40	Название
Cena	Numeric, 10,2	Цена
Kol	Integer	Количество

**3. Ввод данных в таблицы.**

Для добавления записей необходимо выбрать нужную таблицу и нажать на кнопку Browse.

Чтобы добавить пустую запись, нужно выбрать в меню Table команду Append New Record. Затем ввести данные в пустую запись.

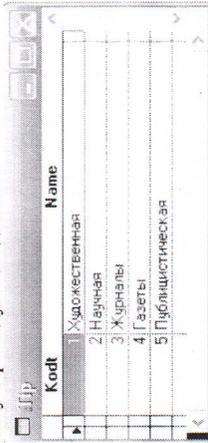
Table Window Help

- Append New Record
- Toggle Deletion Mark
- Font...

- Go to Record
- Append New Record
- Toggle Deletion Mark

Второй способ – набрать команду Append Blank в командном окне.

Введите в таблицу Tip следующие данные:



Введите в таблицу knig следующие данные:

Кодкп	Колкп	Автор	Название	Цена	Кол
1	1	Павлов А. С.	Естественные Очерки	150,00	2
2	2	Есенин С. А.	Вербас	45,00	7
3	3	Мир ПР. И. И.	Мир ПР. И. И.	20,00	5
4	4	Мир Ф. И. И.	Мир Ф. И. И.	55,00	5
5	5	Герасим С.	Мир Ф. И. И.	100,00	3
6	6	Роман Е. В.	Мир Ф. И. И.	90,00	4
7	7	Иванов А. М.	Мир Ф. И. И.	15,00	5
8	8	Иванов А. М.	Мир Ф. И. И.	125,00	1
9	9	Кузнецов А. А.	Древние времена	138,00	2
10	10	Розенберг Е. С.	Закон джордана	255,00	3
11	11	Прокофьев В. И.	Химия	96,00	10
12	12	Левин А. П.	Математический анализ	180,00	12
13	13	Колосовская Г. В.	Колосовская Г. В.	25,00	2

4. Покажите результат преподавателю.
5. Завершите работу приложения.

4. **Выполнение самостоятельного задания.**  
По аналогии с выполненным заданием создайте базу данных предложенного вам варианта.

Заполните созданные таблицы контрольными данными (введите новые записи) - в главной таблице не менее 5 записей, в дочерней - не менее 10.

«Ввод, редактирование и удаление данных»

**Ввод записей в таблицу.**

Откройте в своей папке созданную ранее БД Мудача, которая содержит две таблицы - **Тир** и **Книг**.

Добавьте в таблицу **Тир** новую запись - поле **кодт=6**, поле **name= 'Учебники'**.

Откройте таблицу **Книг**. Добавьте в нее дополнительно две произвольные записи по типу книги **Учебники (кодт=6)**.

1. **Редактирование данных в таблице.**

Перейдите в таблицу **Тир**. Выполните в командном окне команду:

REPLACE name WITH 'Научно-фантастическая' FOR name='Научная'

**Проверьте результат замены данных в таблице - команда BROWSE.**

Аналогично замените данные 'Учебники' на 'Учебники и задачки'.

**Выполните команду и результат скопируйте и вставьте в файл отчета.**

Перейдите в таблицу **Книг**. Выполните в командном окне следующие команды по замене данных:

- заменить поле **пазвант** значением 'Молекулярная химия', в записи со значением этого поля 'Химия';
- **проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

• заменить поле **кол** значением 3 у всех записей со значением 2;

• **проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

• заменить поле **цена** значением 30.00 у всех записей со значением этого поля, меньшим 30;

• **проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

• заменить поле **цена** значением 20.00 у записи со значением поля **пазвант = '168 часов № 55'**;

• **проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

2. **Удаление данных в таблице.**

Перейдите в таблицу **Книг**. Выполните в командном окне команду:

DELETE ALL FOR код<4

**Проверьте результат выполнения команды, просмотрев данные; (дополните отчет). Сколько записей было удалено с помощью этой команды?**

Выполните команду:

RECALL all

**Проверьте результат выполнения команды, просмотрев данные; (дополните отчет). Что произошло в ранее удаленными записями?**

Аналогичным образом выполните следующие команды по удалению данных:

- удалить все записи, у которых значение поля **кодт** равно 1;
- **проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**
- Восстановите удаленные записи;
- удалить все записи, у которых значение поля **кол** равно 5;
- **проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

• Восстановите удаленные записи;

• удалить все записи, у которых значение поля **цена** больше 100;

• **проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

• Восстановите удаленные записи;

• удалить все записи, у которых значение поля **кол** больше 8 и значение поля **цена** меньше 100;

• **проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

• Восстановите все удаленные записи.

3. **Вывод данных на экран дисплея.**

Перейдите в таблицу **Книг**. Установите указатель на запись, у которой поле **кодкп=6**. Выполните в командном окне команду: DISPLAY

**Проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

Выполните в командном окне команду: DISPLAY ALL

**Проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

Используя аналогичным образом команды вывода данных, выполните следующие действия:

• Вывести на экран поля **аутор** и **пазвант** всех записей;

• **проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

• Вывести на экран поля **аутор**, **пазвант** и **кол** всех записей, у которых количество больше 9;

• **проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

• Вывести на экран поля **аутор**, **пазвант** и **цена** всех записей, у которых цена меньше 120;

• **проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

• Вывести на экран поля **аутор**, **пазвант**, **цена** и **кол** всех записей, у которых поле **кодт=6**;

• **проверьте результат выполнения команды, просмотрев данные; (дополните отчет);**

- Вывести на экран структуру текущей таблицы;
- проверьте результат выполнения команды, просмотрев данные: (Дополните отчет).

*Выполните аналогичные действия, используя команду LIST.*

4. Перемещение по записям таблицы. Перейдите в таблицу *Книг*. Используя команды перемещения, выполните следующие действия:
  - переместите с помощью команды указатель на последнюю запись в таблице;
  - проверьте результат выполнения команды, просмотрев данные: (Дополните отчет);
  - проверьте значение функции eof() (для вывода значения на экран *выполните команду ? eof()*); (Дополните отчет);
  - переместите с помощью команды указатель на первую запись в таблице;
  - проверьте результат выполнения команды, просмотрев данные;
  - проверьте значение функции bof() (для вывода значения на экран *выполните команду ? bof()*); (Дополните отчет);
  - переместите с помощью команды указатель на запись с номером 5;
  - проверьте результат выполнения команды, просмотрев данные: (Дополните отчет);
  - переместите с помощью команды указатель на предыдущую запись;
  - проверьте результат выполнения команды, просмотрев данные: (Дополните отчет);
  - переместите с помощью команды указатель на 2 запись вниз;
  - проверьте результат выполнения команды, просмотрев данные: (Дополните отчет);
  - переместите с помощью команды указатель на 4 запись вверх;
  - проверьте результат выполнения команды, просмотрев данные: (Дополните отчет);

5. Покажите файл с результатами преподавателя.

6. Выполните самостоятельного задания.

Откройте базу данных вашего варианта, созданную ранее. Добавьте в таблицы недостающие данные.

Выполните в командном окне тренировочные команды по замене данных, удалению и восстановлению записей, выводу записей на экран, перемещению по записям таблицы.

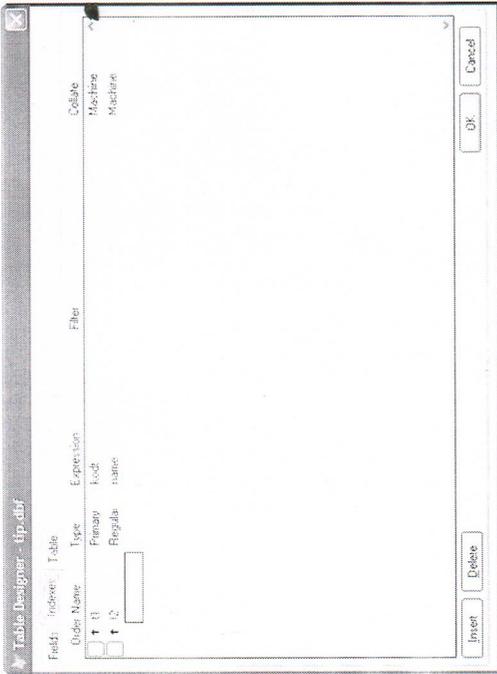
«Создание индексов и определение отношений между таблицами»

1. Создание индексов с помощью конструктора.

Откройте в своей папке созданную ранее БД *Mudata*. Для таблицы *Гир* откройте конструктор таблиц и в окне конструктора перейдите на вкладку *Indexes*. Создайте для данной таблицы два индекса:

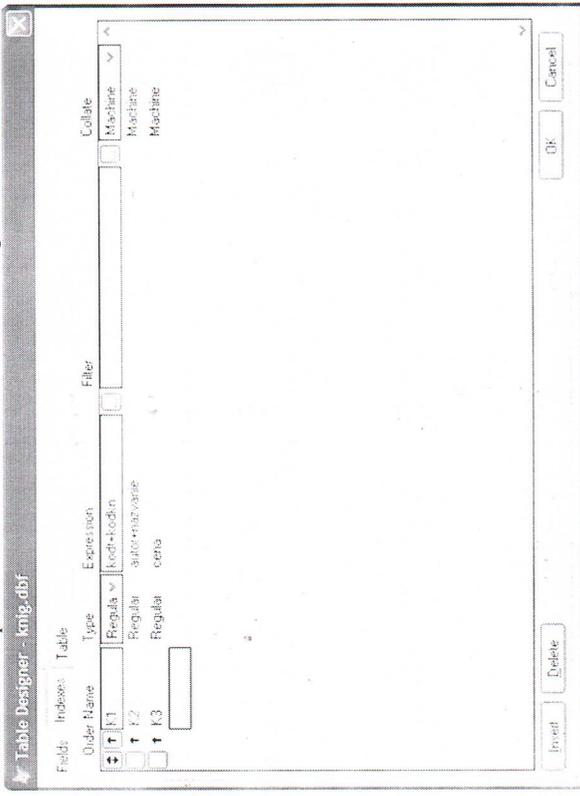
2. Установление отношений между таблицами.

Перейдите в конструктор БД, в котором находятся таблицы *Гир* и *Книг*, с созданными индексами.

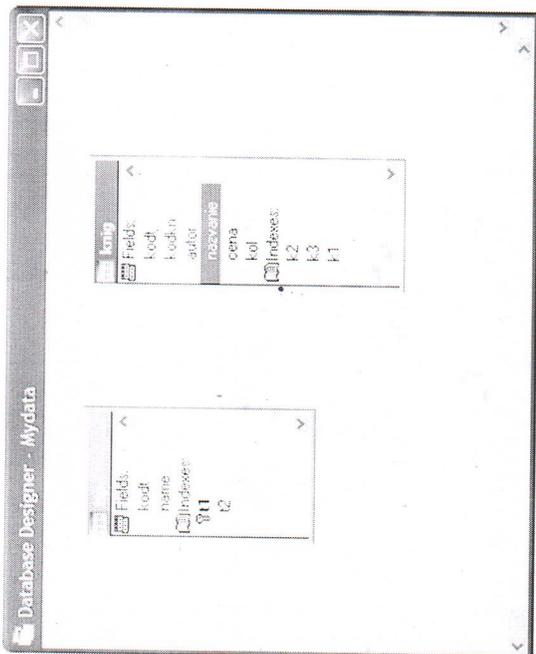


Сохраните внесенные изменения. Обратите внимание, что созданные индексы появились в макете таблицы в конструкторе БД.

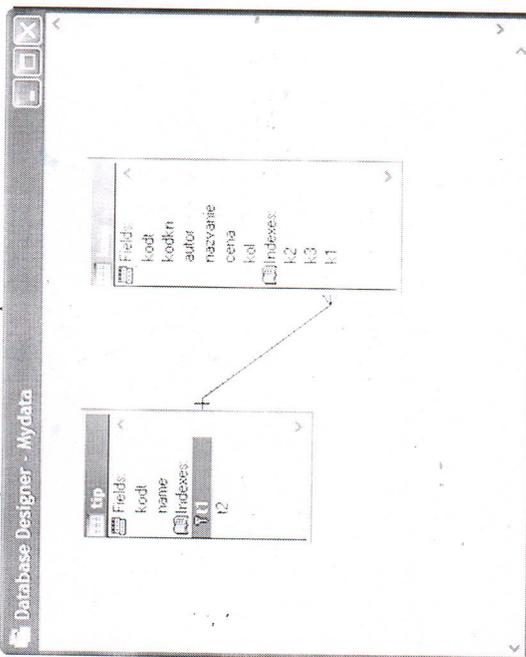
Аналогичным образом создайте индексы для таблицы *Книг*.



2. Установление отношений между таблицами. Перейдите в конструктор БД, в котором находятся таблицы *Гир* и *Книг*, с созданными индексами.



Установите долговременную связь между таблицами, перетащив ключевой индекс главной таблицы на соответствующий индекс дочерней.



### 3. Контроль целостности данных

Для созданной связи выберите пункт Edit Referential Integrity контекстного меню и установите каскадные правила обновления и каскадные правила удаления, а также правила вставки Restrict.

### 4. Выполнение самостоятельного задания

Откройте базу данных вашего варианта, созданную ранее. Создайте для каждой таблицы необходимые индексы. Для главной таблицы должен быть обязательно создан первичный (ключевой) индекс.

Установите долговременные отношения между таблицами, используя созданные индексы. Осуществите контроль целостности данных.

### 5. Покажите результат преподавателю.

«Использование индексов»

#### 1. Создание дополнительных индексов.

Откройте в своей папке созданную ранее БД Mydata. Для таблицы Knig создайте дополнительные индексы:

- Индекс k4 - по полю kodkn;
- Индекс k5 - по полю kol;
- Индекс k6 - по полям autor и kol;

#### 2. Выбор активного индекса в процессе выполнения.

Откройте таблицу Knig. Выполните команды, которые по очереди подключают все индексы (set order to ...).

После подключения очередного индекса, просмотрите результат и добавьте команду и результат в отчет.

#### 3. Обновление индексов.

Откройте таблицу Knig в монопольном режиме. Выполните для данной таблицы переиндексирование.

Аналогично выполните обновление индексов для таблицы Tip.

#### 4. Выполнение самостоятельного задания.

Откройте базу данных вашего варианта, созданную ранее. Создайте для каждой таблицы дополнительные индексы по разным полям.

Выполните подключение индексов в процессе работы и просмотрите результаты.

#### 5. Покажите файл с результатами преподавателю.

**Контрольные вопросы:**

1. Что такое индекс?
2. Опишите этапы создания БД?
3. Перечислите основные элементы БД?

## Практическая работа №10 «Мониторинг работы сервера».

### Тема 1.2 Разработка и администрирование БД

**Цель работы:** «создать, отредактировать запись в таблице»

**Материально-техническое обеспечение:** Компьютер, операционная система Windows 7, FoxPro

#### Краткие теоретические сведения:

##### Порядок выполнения практической работы:

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

#### Задания для выполнения практической работы:

##### Добавление записи в таблицу через команды

1. Нам необходимо в первую очередь, конечно же открыть нашу Базу Данных.
2. Далее нам необходимо создать программу, через которую, будет все работать.
3. Далее необходимо наше «Меню» которое мы сделали в «3 работе по FoxPro». Оно нам пригодится для того, чтобы добавить новый пункт, который нужен в нашей практической работе.

4. Подготовить отчет.

5. Написать вывод о проделанной работе.

Для начала нам нужно открыть нашу Базу Данных:

Для этого вам необходимо написать команду в окне «Command» команду «Browse» и нажать Enter:

```
Command
.DO "c:\users\kent\desktop\3 работа foxpro\porttab1"
.DO program2.prg
.BROWSE
```

Рис.1(Команда Browse)

Далее у вас откроется окно, для выбора вашей Базы Данных:



Рис.2(Выполнение команды «Browse»)

Нажимаем «OK» и видим нашу таблицу с нашими данными:

Name	SecondName	BirthDay	Telephone
Илья	Андреев	04/28/98	9521637603
Владислав	Алексеев	05/25/98	9512589363
Игорь	Парков	12/01/98	9514789566
Геннадий	Шелеев	05/14/98	9521697587
Никита	Бондаренко	04/15/98	9531587987
Владислав	Дубовик	04/01/98	9231234564
Клара	Евгений	07/02/98	9235498798
Корольев	Александр	01/01/98	9236598988
Давид	Никита	02/02/98	9234564567
Кондратов	Геннадий	02/04/98	9231231231
Мариантлев	Дмитрий	04/25/98	9216598788
Ястребов	Анастасия	01/25/98	8945987969

Рис.3(Наша таблица)

Чтобы создать новый файл, необходимо: New -> Program -> New file:

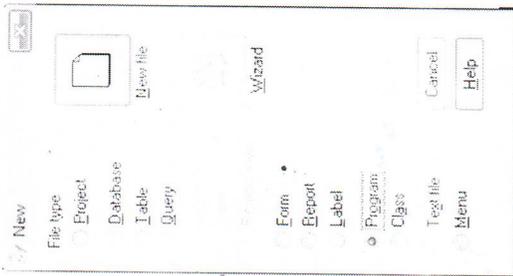


Рис.4(Создание)

После того, как мы создали программу, необходимо вписать эти параметры, у меня она будет называться «sp4».

Команда USE у каждого по-своему, она выбирает ваш путь, к вашей же таблице:



Рис.5(Программ)

Но после этого, не запускайте ее, потому что мы не определили ее в нашем меню.

Для того, чтобы добавить эту программу в само меню, вам необходимо проделать несколько действий.

Для начала зайти в программу, где находится сама меню.

Дайтее добавим новый пункт меню, который будет носить имя «Добавить».

Сначала впишем ее, чтобы она отображалась:

Я добавил ее в выпадающую кнопку «Аудитории»:

```

define bar 1 of v prompt "кабинет 218"
define bar 2 of v prompt "кабинет 115"
define bar 3 of v prompt "кабинет 114"
define bar 4 of v prompt "Добавить"
ON SELECTION PAD F2 OF Andreev activate popup v |

```

Рис.6(Кнопка «Добавить»)

Теперь сделаем, чтобы наша кнопка «Добавить» обращалась к нашей программе «sp4»:

```

on selection bar 1 of v do sp1
on selection bar 2 of v do sp1
on selection bar 3 of v do sp2
on selection bar 4 of v do sp4

```

Рис.7(Добавление программы для работоспособности)

И конечно же, проверим работоспособность, для этого:



Рис.8(Проверка)

После откроется окно, для того, чтобы вы выбрали свою программу:



Рис.9(Выбор программы)

После выбора видим:

Введите ваше имя Владимир

Рис. 10 (Выполнение программы)

Вам надо стереть то, что написано в поле, и ввести то, что нужно вам.

Ввели, и смотрим, что получилось:

Name	Surname	Birthday	Telephone
Илья	Андреев	04/28/98	9521697603
Владислав	Александров	05/25/98	9512586969
Игорь	Паршинов	12/01/98	9514789566
Геннадий	Шелюк	06/14/98	9521697587
Никита	Бондаренко	04/15/98	9531587987
Владислав	Дубовик	04/01/98	9231234564
Харлаев	Евгений	07/03/98	9235498798
Королев	Александр	01/01/98	9236598986
Давыдов	Никита	02/03/98	9234564567
Константинов	Геннадий	02/04/98	9231231231
Климентьев	Дмитрий	04/25/98	9216598788
Ястребов	Андрей	01/25/98	8945987989
		/ /	

Рис. 11 (Проверка работоспособности)

**Контрольные вопросы:**

1. Что такое запись в таблице?
2. Как можно добавить запись в таблицу?
3. Какие параметры можно настроить для записи?

**Практическая работа №11 «Выполнение резервного копирования».**

**Тема 1.3 Организация защиты данных в хранилищах**

**Цель работы:** «получить некоторые практические навыки по созданию меню в программе Visual FoxPro»

**Материально-техническое обеспечение:** Компьютер, операционная система Windows 7, FoxPro

**Краткие теоретические сведения:**

Команды для создания меню:

Define Menu <имя меню> color t/w – Создание меню

DEFINE PAD <имя пункта> OF <имя меню> PROMPT «название» – Создание горизонтального пункта

ON PAD <имя пункта> OF <имя меню> ACTIVATE POPUP <имя меню раскрывающегося> – Активация горизонтального пункта

DEFINE POPUP <имя меню раскрывающегося> – Создание раскрывающегося пункта

DEFINE BAR 1 OF <имя меню раскрывающегося> PROMPT «название» – Создание вертикального пункта

ON SELECTION BAR 1 OF <имя меню раскрывающегося> do <имя программы> – Активация вертикального пункта

**Порядок выполнения практической работы:**

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

**Задания для выполнения практической работы:**

**Создание меню**

1. Создать меню;
2. Создать в меню 3 PAD;
3. Создать в каждом PAD POPUP;
4. Создать в каждом POPUP 3 BAR;
5. 1 PAD и 2 PAD, а также в 3 PAD 1 и 2 BAR могут называться как угодно;
6. В 3 PAD 3 BAR должен обязательно называться.
7. Написать программу для BAR «Выход»;

1. Для начала создадим меню.

```
DEFINE MENU Демон_Меню COLOR T/W
```

Рисунок – 1 Создание меню.

2. После создания меню нужно добавить горизонтальные пункты.

```
DEFINE PAD P1 OF Демон_Меню PROMPT "Земля"
DEFINE PAD P2 OF Демон_Меню PROMPT "Воздух"
DEFINE PAD P3 OF Демон_Меню PROMPT "Настройки"
```

Рисунок – 2 Создание горизонтальных пунктов.

3. Создадим раскрывающийся пункт.

```
DEFINE POPUP POP1
DEFINE POPUP POP2
DEFINE POPUP POP3
```

Рисунок – 3 Создание раскрывающихся пунктов.

4. Теперь нам нужно чтобы при нажатии на горизонтальный пункт появлялся раскрывающийся пункт.

```
ON PAD P1 OF Дакзим_Меню ACTIVATE FORUP FOR1
ON PAD P2 OF Дакзим_Меню ACTIVATE FORUP FOR2
ON PAD P3 OF Дакзим_Меню ACTIVATE FORUP FOR3
```

Рисунок – 4 Активация раскрывающихся пунктов.

5. Далее примемся за создание вертикальных пунктов в каждом раскрывающемся пункте.

```
DEFINE BAR 1 OF FOR1 PROMPT "Окно"
DEFINE BAR 2 OF FOR1 PROMPT "Окно (без рамки)"
DEFINE BAR 3 OF FOR1 PROMPT "Полный экран"
DEFINE BAR 1 OF FOR2 PROMPT "Экспорт"
DEFINE BAR 2 OF FOR2 PROMPT "Импорт"
DEFINE BAR 3 OF FOR2 PROMPT "Открыть"
DEFINE BAR 1 OF FOR3 PROMPT "Инструменты"
DEFINE BAR 2 OF FOR3 PROMPT "Информация"
DEFINE BAR 3 OF FOR3 PROMPT "Выход"
```

Рисунок – 5 Создание вертикальных пунктов.

6. Напишем программу для закрытия нашего меню.

```
ON SELECTION BAR 3 OF FOR3 DO exitprog
PROCEDURE exitprog
* activate window wiped
CLOSE ALL
IN STATE.F.
DEACTIVATE FORUP M_menus
QUIT
```

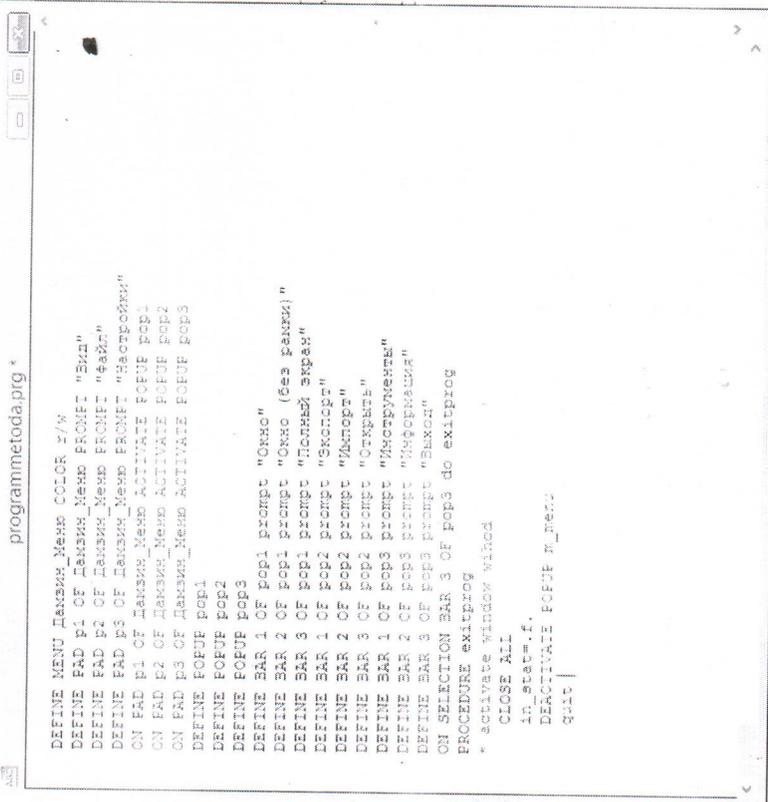
Рисунок – 6 Алгоритм закрытия меню.

7. Теперь нам нужно все это сохранить в файл \*.prg (По умолчанию сохраняется в C:\Users\Имя пользователя\Documents\Visual FoxPro Projects\NA WIN 8.1!!!).

```
SAVE TO ProgramMetoda.prg
DO ProgramMetoda.prg
```

Рисунок – 7 Создание файла \*.prg.

8. Через пункт меню file выбираем пункт open и открываем наш файл \*.prg и в него уже копируем все из командной строки и сохраняем комбинацией клавиш Ctrl+S.



```
programmetoda.prg
DEFINE MENU Дакзим_Меню COLOR 2/1/0
DEFINE PAD P1 OF Дакзим_Меню PROMPT "Меню"
DEFINE PAD P2 OF Дакзим_Меню PROMPT "Файл"
DEFINE PAD P3 OF Дакзим_Меню PROMPT "Настройки"
ON PAD P1 OF Дакзим_Меню ACTIVATE FORUP FOR1
ON PAD P2 OF Дакзим_Меню ACTIVATE FORUP FOR2
ON PAD P3 OF Дакзим_Меню ACTIVATE FORUP FOR3
DEFINE FORUP FOR1
DEFINE FORUP FOR2
DEFINE FORUP FOR3
DEFINE BAR 1 OF FOR1 PROMPT "Окно"
DEFINE BAR 2 OF FOR1 PROMPT "Окно (без рамки)"
DEFINE BAR 3 OF FOR1 PROMPT "Полный экран"
DEFINE BAR 1 OF FOR2 PROMPT "Экспорт"
DEFINE BAR 2 OF FOR2 PROMPT "Импорт"
DEFINE BAR 3 OF FOR2 PROMPT "Открыть"
DEFINE BAR 1 OF FOR3 PROMPT "Инструменты"
DEFINE BAR 2 OF FOR3 PROMPT "Информация"
DEFINE BAR 3 OF FOR3 PROMPT "Выход"
ON SELECTION BAR 3 OF FOR3 DO exitprog
PROCEDURE exitprog
* activate window wiped
CLOSE ALL
IN STATE.F.
DEACTIVATE FORUP M_menus
QUIT
```

Рисунок – 8 Общий результат работы.

Чтобы проверить работоспособность программы нужно открыть пункт меню Program в нём выбрать Do... далее выбираем нашу программу и пишем АСТВИТАЕ MENU <Название меню>.

**Контрольные вопросы:**

1. Что такое меню?
2. Перечислите этапы разработки меню в FoxPro?

### Практическая работа №12 «Восстановление базы данных из резервной копии»

#### Тема 1.3 Организация защиты данных в хранилищах

**Цель работы:** «Создать в программе меню дополнительные пункты в меню которые будут открывать новые окна с информацией о разработке и базой данных»

**Материально-техническое обеспечение:** Компьютер, операционная система Windows 7, FoxPro

#### Краткие теоретические сведения:

Добавление новых пунктов:

```
DEFINE BAR 6 OF M_MENU PROMPT "Информ"
DEFINE BAR 7 OF M_MENU PROMPT "Средства"
```

```
ON SELECTION BAR 6 OF M_MENU DO Inform
ON SELECTION BAR 7 OF M_MENU DO sredstva
```

#### Порядок выполнения практической работы:

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

#### Задания для выполнения практической работы:

##### Создание дополнительных окон в Visual FoxPro

1. Добавить в наше меню новые пункты

```
DEFINE BAR 6 OF M_MENU PROMPT "Информ"
```

```
DEFINE BAR 7 OF M_MENU PROMPT "Средства"
```

```
ON SELECTION BAR 6 OF M_MENU DO Inform
```

```
ON SELECTION BAR 7 OF M_MENU DO sredstva
```

##### Рисунок 1

2. Добавляем в меню команды выполнения наших программ для новых окон (О создателе и база данных)

```
PROCEDURE Inform
DO program_test.prg
```

```
PROCEDURE sredstva
DO sredstva.prg
```

##### Рисунок 2

3. Пишем код программы для открытия базы данных в новом окне.

```
DEACTIVATE WINDOW all
```

```
RELEASE WINDOWS all
```

```
USE "c:\as\bazazh.cbf" SHARED
```

```
DEFINE WINDOW f2 FROM 1,1 TO 10,120 COLOR 1/16 FONT 'arial'
```

```
ACTIVATE WINDOW f2
BROWSE
```

Рисунок 3

4. Пишем код для программы о создателе.

(Если задать шрифт стандартный, то русский текст не будет отображаться корректно)

```
DEFINE WINDOW f3 FROM 12,1 TO 15,35 COLOR 1/16 FONT 'arial'
```

```
ACTIVATE WINDOW f3
```

```
Ø1,1 say 'Created by Absolutely Sempranu'
```

Рисунок 4

Проверяем работоспособность наших пунктов в меню



Рисунок 5

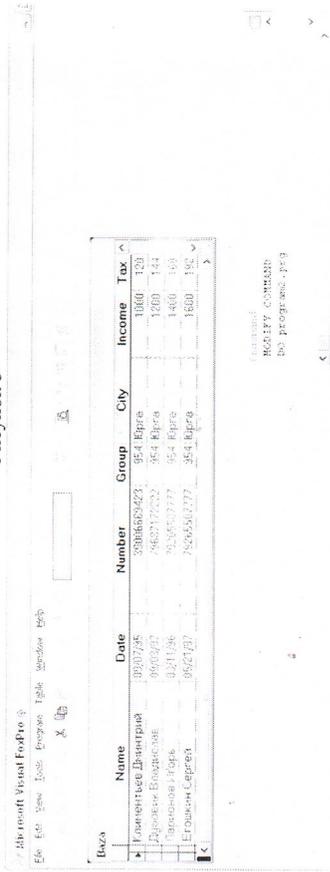


Рисунок 6

#### Контрольные вопросы:

1. Что такое окно программы?
2. Какие возможности по работе с окном дает FoxPro?

Практическая работа №13 «Реализация доступа пользователей к базе данных».

### Тема 1.3 Организация защиты данных в хранилищах

**Цель работы:** «Преобразование реляционной БД»  
**Материально-техническое обеспечение:** Компьютер, операционная система Windows 7, FoxPro

#### Порядок выполнения практической работы:

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

#### Задания для выполнения практической работы:

##### 1. Создание проекта. Исследование созданного проекта.

Создайте проект любым из имеющихся способов (посредством главного меню или с помощью команды). При сохранении введите имя проекта:

**ВНИМАНИЕ!** Имя вашего проекта – Мурroj\_N, где N – ваш номер по журналу (например, если ваш номер 3, то имя проекта – Мурroj\_3).

Изучите разделы проекта и просмотрите доступные функциональные кнопки проекта.

##### 2. Добавление файлов в проект.

###### Задание 1.

Перейдите на вкладку **Data** и выберите в списке пункт **Database**. Добавьте в проект базу данных **Murdata**, созданную ранее. Обратите внимание, что при установке курсора на добавленную БД, в нижней части проекта отображается полный путь к файлу БД. **Убедитесь, что путь соответствует вашей папке!**

Проверьте работу функциональных кнопок на данной вкладке.

На вкладке **Data** выберите пункт **Free Tables** (свободные таблицы) и добавьте в этот раздел свободную таблицу вашего варианта, созданную на первом практическом занятии.

**Убедитесь, что путь таблицы соответствует вашей папке!** Проверьте работу функциональных кнопок при работе со свободными таблицами.

Скопируйте скрин-шот с вкладки **Data** в файл отчета.

###### Задание 2.

Перейдите на вкладку **Documents**. Изучите активные функциональные кнопки.

###### Задание 3.

Перейдите на вкладку **Code**. Изучите активные функциональные кнопки. Выберите в списке пункт **Program**. Добавьте все разработанные вами программы в проект.

**Убедитесь, что пути программных файлов соответствуют вашей папке!** Проверьте работу функциональных кнопок при работе с программными файлами. Используйте кнопки, откройте любую программу для изменения, попробуйте запустить программу и т.д.

Используя кнопки проекта, создайте новую программу **prog20**, которая решает задачу, выданного вам варианта (на карточке).

**Проверьте наличие созданной программы и результат ее выполнения добавьте в отчет. Самостоятельно добавьте ее в проект. Скопируйте скрин-шот с вкладки Code в файл отчета.**

###### Задание 4.

Перейдите на вкладку **Other**. Изучите активные функциональные кнопки. Выберите в списке пункт **Other Files**. Добавьте в проект несколько картинок (файлы с расширением .bmp, .jpg и т.д.).

Скопируйте скрин-шот с вкладкой **Other** в файл отчета.

3. **Покажите результат работ преподавателю. Закройте созданный проект. После выхода в операционную систему проверьте наличие в вашей папке файлов с вашим именем Мурroj\_N.jpg и Мурroj\_N.pjt.**

«Создание запросов»

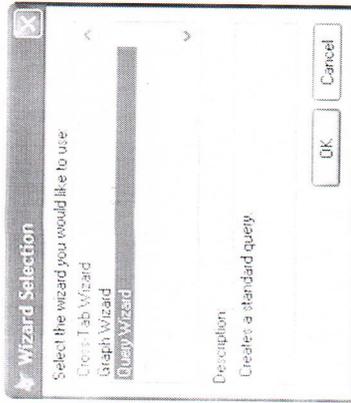
Для возможности проверки преподавателем ваших выполненных заданий, необходимо создать файл отчета в среде Microsoft Word, в который скопировать все выполненные команды и скрин-шоты с результатами.

#### Откройте свой проект, созданный ранее.

##### 1. Создание запроса с помощью мастера из главного меню.

**Задание:** создать запрос на основе таблицы **knig**, содержащий поля **Название** и **Автор**.

В главном меню выберите пункт **File – New** и в окне **New** выберите создание запроса с помощью мастера. В окне **Wizard Selection** выберите мастер запросов:



Выполните шаги мастера:

1 шаг – из таблицы **Knig** выберите поля **пazvane** и **autor**.

2 шаг – здесь отсутствует, так как запрос создается на основе одной таблицы.

3 шаг – оставьте без изменения.

4 шаг – установите сортировку по полю **пazvane** по возрастанию.

5 шаг – завершите создание запроса.

Сохраните запрос под именем **knig1**. Проверьте наличие созданного запроса в своей папке.

2. **Запуск запроса на выполнение разными способами.**

В командном окне выполните команду **DO knig1.qpr** и посмотрите созданный запрос.

**Добавьте в свой отчет скрин-шот с командой и результатом ее выполнения.**

Do. Затем попробуйте запустить запрос, используя опции главного меню – Program –

**Data** и выберите раздел **QUERIES**, нажмите кнопку **Add**, выберите имя созданного вами запроса и нажмите на кнопку **Ok**.

Запустите запрос из проекта – кнопка **Run**.

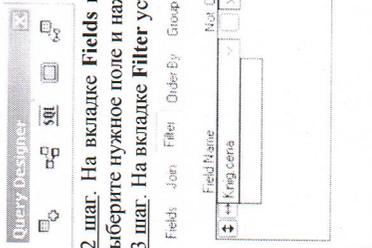
3. **Создание запроса с помощью команды CREATE QUERY в режиме конструктора.**

**Задание:** создать запрос на основе таблицы **knig**, содержащий поля **Название, Автор и Цена** и запись с ценой больше 100.

В командном окне выполните команду **CREATE QUERY**. В результате запускается конструктор запросов.

1 шаг. Выберите таблицу, на основе которой будет создан запрос – **knig**, нажмите кнопку **Add** и закройте окно **Add Table or View**.

Для удобства работы в конструкторе запросов используется панель инструментов



2 шаг. На вкладке **Fields** выберите поля, включаемые в запрос (см. задание). Для этого выберите нужное поле и нажмите кнопку **Add**.

3 шаг. На вкладке **Filter** установите условие фильтра:

Для создания используйте разворачивающийся список.

4 шаг. На вкладке **Order By** выберите условие сортировки – по полям **autor** и **nazvanie**.

5 шаг. Для просмотра запроса нажмите кнопку **Preview** на панели инструментов.

**Добавьте в свой отчет скрин-шот с результатом выполнения запроса.**

6 шаг. С помощью кнопки **Print** панели инструментов просмотрите текст программы, которая создает запрос.

**Добавьте в свой отчет скрин-шот с программным кодом.**

7 шаг. Закройте окно конструктора запросов. Сохраните запрос под именем **knig\_sep**. Добавьте созданный запрос в свой проект.

Добавьте созданный запрос любым способом и просмотрите результат.

**Добавьте в свой отчет скрин-шот с результатом выполнения запроса.**

4. **Изменение существующего запроса.**

Выберите в проекте имя созданного вами запроса **knig\_sep** и нажмите на кнопку **Modify**. В результате открывается конструктор запросов.

На вкладке **Filter** измените условие отбора записей: **cena < 100**. Сохраните запрос и просмотрите результат.

**Добавьте в свой отчет скрин-шот с результатом выполнения запроса.**

5. **Создание запроса из проекта с помощью конструктора.**

Перейдите в проекте на вкладку **Data** и выберите раздел **QUERIES**, нажмите кнопку **New**. В появившемся окне нажмите кнопку **New Query**.

Далее аналогично предыдущему запросу создайте запрос: на основе таблицы **Tip**. В запросе выберите поле **name** и установите сортировку по этому же полю. Сохраните запрос с именем **tip1** и просмотрите его.

**Добавьте в свой отчет скрин-шот с результатом выполнения запроса.**

6. **Создание запроса на основе двух таблиц с помощью мастера.**

**Задание:** создать запрос на основе двух таблиц – **tip** и **knig**, содержащий поля **Наименование типа, Название книги, Автор и Цена**.

Перейдите в проекте на вкладку **Data** и выберите раздел **QUERIES**, нажмите кнопку **New**. В появившемся окне нажмите кнопку **Query Wizard**. В результате будет запущен мастер создания запросов.

1 шаг – Выбор полей. Из таблицы **Tip** выберите поле **name**, из таблицы **Knig** выберите поля **Автор, Название, Цена**.

2 шаг – Установление связи между таблицами. Нажмите кнопку **Add** для выбора связи.

2а шаг – Записи, включаемые в запрос. Оставляем без изменения.

3 шаг – Фильтрация записей. Для эксперимента установите фильтр для таблицы **Tip**

Field: **Tip.kodt** Operator: **equals** Value: **1**

и нажмите кнопку **Preview**, чтобы посмотреть результат. В запросе содержатся только записи, у которых **kodt** равен 1.

Очистите условие фильтра и перейдите на следующий шаг.

4 шаг – Сортировка записей. Установите сортировку по полю **name**.

4а шаг – Лимит записей. Оставляем без изменения.

5 шаг – Финиш. Нажмите кнопку **Finish** и сохраните запрос с именем **tip\_kn**.

Запустите запрос для просмотра. Обратите внимание, что в нем содержатся данные из двух таблиц, что очень удобно для просмотра.

**Добавьте в свой отчет скрин-шот с результатом выполнения запроса.**

Нажмите в проекте кнопку **Modify** для созданного запроса и перейдите на вкладку **Join**. Проверьте, правильно ли установлена связь между таблицами.

7. **Покажите результат преподавателю. Закройте свой проект и осуществите выход из системы.**

**Контрольные вопросы:**

1. Что такое проект в FoxPro?
2. Какие виды проектов существуют?
3. Каким способом можно создать проект в БД?

Практическая работа №14 «Мониторинг безопасности работы с базами данных».

### Тема 1.3 Организация защиты данных в хранилищах

Цель работы: «Преобразование реляционной БД»

Материально-техническое обеспечение: Компьютер, операционная система Windows 7, FoxPro

Краткие теоретические сведения:

Порядок выполнения практической работы:

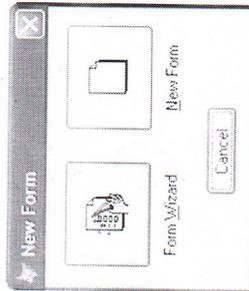
1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

Задания для выполнения практической работы:  
«Создание форм с помощью мастера»

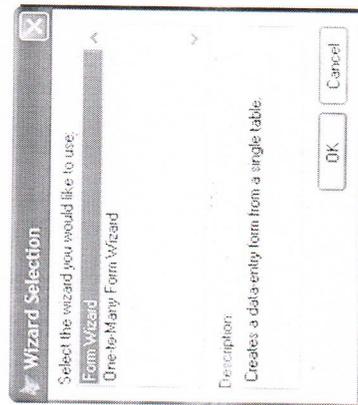
Откройте свой проект, созданный ранее.

1. Создание однотабличной формы.

Задание: создать с помощью мастера форму на основе таблицы **izd**.  
Перейдите в проекте на вкладку **Documents** и выберите раздел **Forms**. Для создания новой формы нажмите кнопку **New** в проекте. В появившемся окне **New Form** нажмите кнопку **Form Wizard**.



В появившемся окне **Wizard Selection** выберите пункт **Form Wizard** для создания однотабличной формы.



Выполните шаги мастера, выполняя все его инструкции.

1 шаг – для таблицы **izd** выберите все поля.  
2 шаг – в качестве оформления формы выберите стиль **Shadowed** и тип кнопок **Custom**.

3 шаг – установите сортировку по полю **izd** по возрастанию.

4 шаг – введите заголовок формы – **Данные об издательствах**, просмотрите результат с помощью кнопки **Preview**. Перед сохранением выберите в окне опцию **Save and run form** (Сохранить и запустить форму).

- Save form for later use
- Save and run form
- Save form and modify it in the Form Designer

После этого нажмите кнопку **Finish**.

В появившемся окне **Save As** укажите путь к своей папке и задайте имя новой формы – **izd1**.

Если все действия были выполнены верно, то форма будет сохранена и запущена. Проверьте как осуществляется вывод данных в форме, используя полосу прокрутки. Проверьте работу кнопок.

Добавьте в таблицу новую запись с данными: код издательства – 1, название издательства – Форум, адрес – Москва, фно руководителя – Кузьминчев А.С.

2. Модификация формы.

Задание: внести изменения в форму **izd1**, используя конструктор.

Откройте созданную форму **izd1** в режиме конструктора.

- изменение цвета объектов формы:

Активируйте панель инструментов **Color Palette** с помощью пункта меню **View – издательствах** и цвет фона формы. Запустите форму, сохраните ее и проверьте результат внесенных изменений.

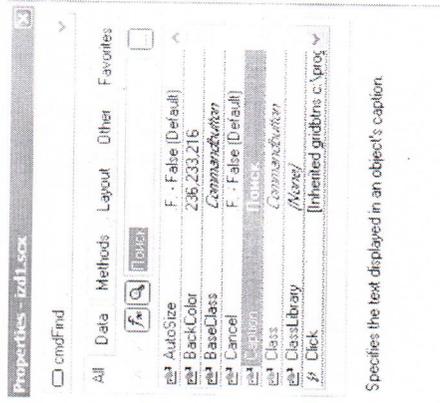
- изменение расположения и размеров объектов формы:

Переместите надпись «Данные об издательствах» так, чтобы она располагалась по горизонтали по центру.

Уменьшите размер контейнера, в котором отображаются данные. Запустите форму и проверьте результат внесенных изменений.

- изменение свойств объектов:

В окне **Properties** выберите в списке объектов формы объект **cmdFind**. Затем в списке свойств этого объекта найдите свойство **Caption** и введите в строке ввода слово **Поиск**.



В результате в форме эта кнопка будет иметь название «Поиск». Аналогично измените названия остальных кнопок: **cmdPrint** - Печать, **cmdAdd** - Ввод, **cmdDelete** - Удаление, **cmdExit** - Выход.

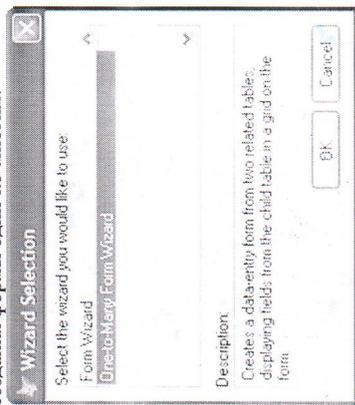
Для кнопки Выход установите свойство **FontBold** равным **.T.**, свойство **ForeColor** равным **255,0,0**. Обратите внимание, что значения всех свойств, которые мы изменяли в окне **Properties**, изменяют свой цвет на фиолетовый.

Запустите форму и проверьте результат внесенных изменений. Закройте форму и сохраните все изменения.

### 3. Создание формы на основе двух таблиц.

**Задача:** создать с помощью мастера форму на основе таблиц *tip* и *knig*.

Запустите мастер форм из проекта. В окне **Wizard Selection** выберите пункт **One-to-Many Form Wizard** для создания формы один-ко-многим:



Выполните шаги мастера:

1 шаг – для родительской таблицы **tip** выберите поле **name**.

2 шаг – для дочерней таблицы **knig** выберите поля **autor**, **nazvanie**, **сena**, **kol**.

3 шаг – проверьте, правильно ли установлена связь между таблицами – должна быть установлена связь по полю **код**.

4 шаг – в качестве оформления формы выберите стиль **Embossed** и тип кнопок **Picture buttons**.

5 шаг – установите сортировку по полю **Name**.

6 шаг – введите заголовки формы – **Перечень книг**, просмотрите результат с помощью кнопки **Preview**.

После этого нажмите кнопку **Finish**.

В появившемся окне **Save As** укажите путь к своей папке и задайте имя новой формы – **tip\_knig**.

### 4. Тестируйте формы.

**Задача:** запустить форму **tip\_knig** и проверить ее работу.

Запустите форму из проекта. Проверьте работу всех кнопок. Обратите внимание на то, какие недочеты имеет форма, чтобы в дальнейшем их устранить.

### 5. Модификация формы.

**Задача:** внести изменения в форму **tip\_knig**, используя конструктор.

Откройте созданную форму **tip\_knig** в режиме конструктора. Откройте все необходимые для работы панели инструментов.

### - изменение размеров объектов:

Так как данные таблицы не полностью видны в форме, то необходимо изменить размер формы (увеличить по ширине) и размер объекта **Grid1** так чтобы отображались все данные. С помощью мыши измените размеры указанных объектов.

Запустите форму, сохраните ее и проверьте результат внесенных изменений.

### - изменение свойств объектов:

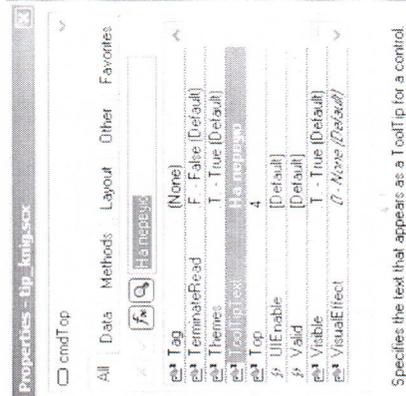
В окне **Properties** выберите в списке объектов формы объект **Form1**. Затем в списке свойств этого объекта найдите свойство **MaxButton** и измените его значение на **.F.** Аналогично для свойства **MinButton** установите значение **.F.** Обратите внимание, что в заголовке формы исчезли кнопки максимизации и минимизации окна.

Запустите форму, сохраните ее и проверьте результат внесенных изменений.

Для надписи **Перечень книг** измените цвет текста и расположите ее по горизонтали по центру. Для текстового поля **Name** и подписи к нему измените размер шрифта и цвет текста.

При запуске формы обратите внимание, что для каждой кнопки в форме имеется всплывающая подсказка, но на английском языке. Это не очень удобно.

В окне **Properties** выберите в списке объектов формы объект **cmdTop**. Затем в списке свойств этого объекта найдите свойство **ToolTipText** и введите в строке ввода слово **На первую**.



Запустите форму и проверьте измененную подсказку. Аналогично измените текст подсказки для всех остальных кнопок – **Предыдущая**, **Следующая**, **На последнюю**, **Поиск**, **Печать**, **Ввод/Сохранение**, **Редактирование**, **Удаление**, **Выход**.

Запустите форму, сохраните ее и проверьте результат внесенных изменений.

Завершите работу с формой **tip\_knig**.

### 6. Покажите результат работы преподавателю.

«Конструктор форм. Настройка параметров формы. Определение среды окружения»

#### 1. Создание экранной формы с помощью команды **CREATE FORM**.

**Задача:** создать форму, настроить основные свойства формы.

В командном окне выполните команду **CREATE FORM**. В результате запускается конструктор форм и открывается окно со свойствами формы.

Для работы с конструктором форм необходимо окно свойств **Properties**, панели инструментов **Form Designer** и **Form Controls**.

## 2. Определение среды окружения формы и вставка в форму элемента «Сетка».

**Задание:** подключить к форме таблицу *tip*, добавить в форму элемент *Grid*, настроить необходимые свойства.

Для подключения данных к форме используется диалоговое окно **Data Environment**. Оно отображает элементы среды данных, с которой работает экранная форма. Для активации окна **Data Environment** используются следующие способы: через главное меню **View**, через контекстное меню, через панель инструментов **Form Designer** – кнопка 

Используйте любой из предложенных способов – на экране появится окно для выбора нужной таблицы. Выберите таблицу **tip** и закройте окно. Выбранная таблица будет вставлена в окно **Data Environment**.

Выделите таблицу и установите для нее свойство **Order** – индекс **1**. Данное свойство подключает индекс для сортировки таблицы.

Для вставки таблицы в форму достаточно перетащить таблицу за заголовок мышкой на форму. Выполните это. В результате в форме создается элемент управления **Grid**. После этого закройте окно **Data Environment**, так как оно нам пока не требуется.

Рассмотрите в окне свойств **Properties** свойства элемента **Grid**, которые установлены по умолчанию (они выделены фиолетовым цветом).

RecordSource	Задает источник данных, к которому привязан элемент управления Grid.
--------------	--

### Установите дополнительные свойства:

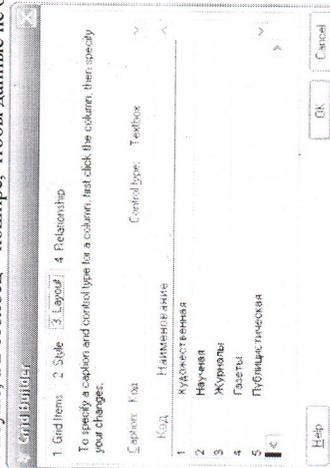
DeleteMark - f.	Определяет, будет ли в элементе управления Grid отображаться столбец маркеров удаления.
ReadOnly - t.	Определяет, может ли пользователь редактировать элемент управления или разрешены ли обновления таблицы или представления, связанных с объектом Cursor. t. - Пользователь не может редактировать элемент управления.

Для настройки элемента **Grid** можно щелкнуть по этому элементу правой кнопкой мыши и использовать команду **Builder...** (построитель) из контекстного меню. Выберите эту команду.

На вкладке **Grid Items** указаны выбранные поля.

На вкладке **Style** выберите по желанию стиль оформления элемента **Grid**.

На вкладке **Layout** в разделе **Caption** введите заголовки столбцов таблицы: 1 столбец – **Код**, 2 столбец – **Наименование**. Здесь же можно изменить ширину столбцов. 1 столбец сделайте поуже, а 2 столбец – пошире, чтобы данные не обрезались.



**Окно свойств**  
Окно **Properties** открывается со списком свойств и событий выделенного объекта и используется для установления его свойств.

**Панель Form Designer**  
Панель инструментов **Form Designer** автоматически отображается вместе с конструктором форм.

**Панель Form Controls**  
С помощью панели инструментов **Form Controls** можно создавать элементы управления в формах.

Если окно свойств отсутствует, то нажмите кнопку  на главной панели инструментов.  
Если панели инструментов отсутствуют, то для их активации выберите пункт меню **View - Toolbars** и выберите нужную панель.

### В окне свойств установите основные свойства формы:

AutoCenter - t.	центрирование объекта Form
BorderStyle - 1	стиль обрамления объекта
Caption - «Типы книг»	текст названия объекта
MaxButton - f.	Кнопка максимизации объекта
MinButton - f.	Кнопка минимизации объекта
Picture	Установка фоновой рисунка
WindowState - 0	0 – немодальный, меню активно. 1 - модальный. В этом случае будет активна только данная форма, никакой другой объект Form не может стать активным, меню неактивно.

Запустите форму для проверки установленных свойств.

Для запуска формы используется кнопка  на главной панели инструментов.

На вкладке **Relationship** устанавливаются связи между таблицами. Так как у нас в форме будет отображаться одна таблица, то здесь ничего не делаем.

После установленных изменений нажмите кнопку **Ok**.  
Измените размеры элемента **Grid** – уменьшите ширину так, чтобы были видны только столбцы таблицы. Разместите таблицу в центре формы.

Запустите форму для просмотра и проверьте внесенные изменения.

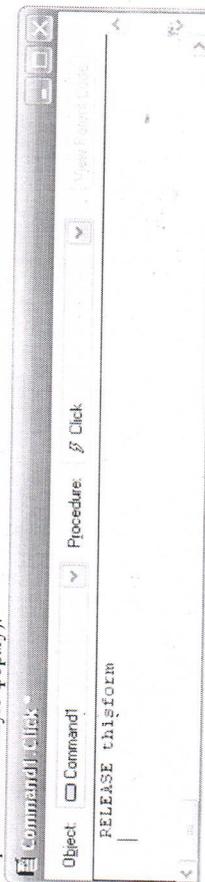
3. **Вставка в форму элемента «Кнопка».**  
*Задание: создать в форме кнопку «Выход», при нажатии которой форма будет закрываться.*

Чтобы вставить элемент Кнопка, щелкните на изображении кнопки  на панели **Form Controls (Command Button)**, затем установите указатель мыши в форме и щелкните для вставки элемента управления. В форме появится элемент кнопки с именем **Command1**.  
Для элемента Кнопка установите следующие свойства:

Caption – «Выход»	Название кнопки
FontSize – 12	Задаёт размер шрифта для текста, отображаемого вместе с объектом
ForeColor – выбрать красный цвет	Задаёт основной цвет символов, используемые при отображении текста и графики в объекте.
FontBold – t.	Полужирное начертание букв
Cancel – t.	Определяет, является ли элемент управления CommandButton кнопкой Cancel, это означает, что если пользователь нажмет клавишу ESC, произойдет событие Click для кнопки Cancel

Для того, чтобы кнопка работала в форме, необходимо обработать событие, которое будет происходить при нажатии кнопки. В кнопках всегда обрабатывается событие **Click** – однократное нажатие левой кнопки мыши.

Щелкните два раза по кнопке **Command1**. Откроется окно **Command1.Click** для ввода обрабатываемой процедуры. Введите одну команду - **release thisform** (эта команда закрывает активную форму).



Закройте окно **Command1.Click**.

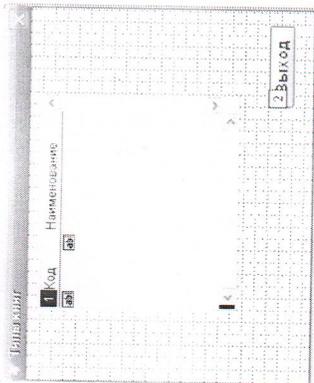
Запустите форму и проверьте работу созданной кнопки.

4. **Настройка порядка обхода элементов управления в форме.**

*Задание: проверить и настроить порядок обхода элементов в форме.*

Для настройки порядка обхода элементов управления в форме используется кнопка **Set Tab Order**  на панели инструментов **Form Designer**.

Нажмите эту кнопку и установите порядок элементов: сетка – 1, кнопка – 2.



Нажмите кнопку повторно. Порядок обхода сохранится. Запустите форму для просмотра.

Как видите, при запуске формы сразу активна таблица и по ней можно перемещаться с помощью стрелок вверх и вниз. Для выхода из формы достаточно нажать кнопку **Выход**.

### 5. **Добавление в форму второй кнопки.**

*Задание: добавить в созданную форму кнопку «Ввод новой записи», при нажатии на которую будет загружаться новая форма – для ввода новой записи в таблицу.*

Добавьте в форму новую кнопку - **Command2**. Установите нужные свойства:

Caption – «Ввод новой записи».

Двойным щелчком мыши вызовите окно **Command2.Click**. Введите следующие команды:

```
do FORM f2 && запуск новой формы
thisform.grd1ip.SetFocus && установка фокуса на элемент Сетка
thisform.Refresh && обновление формы
```

Здесь после знака **&&** введены комментарии. Закройте окно **Command2.Click**.

Созданная кнопка пока не будет работать, так как у нас не создана форма **f2**. Если попробовать запустить форму и нажать данную кнопку то выведет ошибка.

Сохраните форму. Закройте конструктор форм.

Созданную форму необходимо добавить в проект. Перейдите в вкладку **Documents** и выберите пункт **Forms**. Нажмите кнопку **Add** и добавьте созданную форму в свой проект.

### 6. **Создание экранной формы из проекта.**

*Задание: создать форму, установить основные свойства, подключить данные, создать коды обработки событий.*

В проекте перейдите на вкладку **Documents** и выберите пункт **Forms**. Нажмите кнопку **New - New Form**.

Задайте основные свойства формы:

```
AutoCenter – t.
BorderStyle – 1
Caption – название формы «Ввод новой записи»
MaxButton – f.
MinButton – f.
WindowState – 1
```

Подключите к форме таблицу **tip** с помощью окна **Data Environment**. Выделите таблицу и установите для нее свойство **Order** – индекс **t1**. Перетащите в форму из окна **Data Environment** из таблицы **tip** поле **name**. В результате в форму будут вставлены два элемента – **Label** (текст) и **Text Box** (поле для ввода).

Необходимо отредактировать данные элементы.

Выделите элемент **lblName** и измените для него свойство **Caption** – Наименование. Передвиньте элемент так, чтобы он не закрывал поле для ввода.

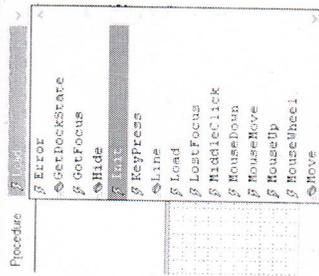
Выделите элемент txtName и измените для него свойство ControlSource – m.name. Это свойство задает источник данных, к которому привязывается объект. В нашем случае осуществляется привязка к переменной памяти, которая будет создана командой SCATTER MEMVAR BLANK.

**Справка:**

Команда SCATTER MEMVAR - копирует содержимое текущей записи таблицы в набор переменных памяти. При ссылке на переменную памяти, чтобы отличать ее имя от имени соответствующего поля в таблице, ставьте перед именем переменной префикс 'm.'. Команда SCATTER MEMVAR BLANK - Создает набор пустых переменных памяти.

Для данной формы необходимо создать код для обработки события загрузки формы – это событие **Init**. Чтобы мы могли ввести новую запись в таблицу **tr**, необходимо создать пустой набор переменных памяти при загрузке формы.

Выполните двойной щелчок мыши по любому свободному месту формы. Откроется окно для обработки кодов формы – **Form1.Load**. Выберите в разделе **Procedure** из разворачивающегося списка событие **Init**:



Обратите внимание, что у окна поменяется название на **Form1. Init**.

Введите команду **SCATTER MEMVAR blank** и закройте окно.

**Сохраните форму с именем f2.**

**7. Вставка в форму кнопок.**

*Задача:* добавить в форму кнопки «Выход» и «Запись».

Добавьте в форму кнопку «Выход» аналогично первой форме.

Добавьте в форму кнопку «Запись», при нажатии которой будет осуществляться запись новых данных в таблицу. Поэтому для данной кнопки нужно ввести код для события **Click**:

```
APPEND BLANK
```

```
REPLACE nam WITH thisform.TxtName.Value
```

```
RELEASE thisform
```

Настройте порядок обхода элементов управления в данной форме: элемент **txtName** – 1, кнопка **Запись** – 2, кнопка **Выход** - 3.

Сохраните форму.

Для проверки работы созданных форм запустите форму **f1** (кнопка **Run** проекта), в ней нажмите кнопку «Ввод новой записи». В появившемся окне введите произвольную новую запись.

**8. Покажите результат работы преподавателю.**

**9. Ответьте на контрольные вопросы.**

**Контрольные вопросы:**

1. Для создания формы используется команда ...

2. Окно Properties используется для ...
3. Для создания среды окружения формы используется ...
4. Панель инструментов Form Controls используется для ...
5. Событие Click означает - ...
6. Событие Init означает - ...
7. Элемент Grid используется для ...
8. Элемент Label используется для ...
9. Элемент TextBox используется для ...

## Практическая работа №15 «Установка приоритетов»

### Тема 1.3 Организация защиты данных в хранилищах

Цель работы: «Преобразование реляционной БД»

Материально-техническое обеспечение: Компьютер, операционная система Windows 7, FoxPro

#### Краткие теоретические сведения:

#### Порядок выполнения практической работы:

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

#### Задания для выполнения практической работы:

##### «Использование стандартных функций Visual Foxpro в программировании»

#### 1. Использование математических функций.

Создайте программу prog16. В программе выполните следующие действия: Переменной x присвойте произвольное вещественное значение с числом десятичных знаков, равное 5 (например, 123.67953).

Используя встроенные математические функции, вычислить:

- Синус числа x;
- Целую часть числа x;
- Результат округления числа x до целого;
- Арктангенс числа x;
- Результат округления числа x до двух десятичных знаков;
- Корень квадратный из числа x;
- Число x в кубе.

Все результаты вывести на экран.

#### 2. Использование символьных функций.

Создайте программу prog17. В программе выполните следующие действия: переменной str присвойте произвольное строковое значение, длиной не менее 20 символов (например, 'Техническое задание').

Используя встроенные символьные функции, выполнить:

- Удалить все начальные и конечные пробелы из строки;
- Выделить из начала строки 5 символов;
- Определить номер позиции, с которой начинается подстрока 'a';
- Преобразовать все буквы в заглавные;
- Выделить из строки 5 символов, начиная с третьего;
- Вычислить длину строки.

Все результаты вывести на экран.

#### 3. Использование функций по обработке даты.

Создайте программу prog18. В программе выполните следующие действия: переменной dd присвойте значение системной даты.

Используя встроенные функции, выполнить:

- Выделить номер дня из даты;
- Выделить номер месяца из даты;
- Выделить номер года из даты;

Все результаты вывести на экран.

#### 4. Покажите результат преподавателю.

#### Контрольные вопросы:

1. Перечислите типы данных в БД.

## Практическая работа №16 «Развертывание контроллеров домена».

Тема 1.3 Организация защиты данных в хранилищах

Цель работы: «Преобразование реляционной БД»

Материально-техническое обеспечение: Компьютер, операционная система Windows 7, FoxPro

### Порядок выполнения практической работы:

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

### Задания для выполнения практической работы: «Компильция проекта приложения»

#### Откройте свой проект, созданный ранее.

1. **Создание главного файла приложения.**
  - Загрузите свой проект и перейдите на вкладку **Code**. Выберите пункт **Program** и нажмите кнопку **New**. У вас откроется окно новой программы.
  - Введите текст программы:

```
set talk off
set safety off
set help off
set memowidth to 90
set repro to auto
set cent on
set date to dmy
set delete on
set point to ''
set near on
clear macros
set excl off
set esca off
set refre to 10
ZOOM WIND SCREEN MAX
MODI WIND SCREEN;
FONT "Courier New Cyr", 12 STYLE "B";
TITLE "Б И Б Л И О Т Е К А";
NOCLOSE NOFLOAT NOZOOM FILL FILE r1.jpg
do menu1.mpr
read events
```

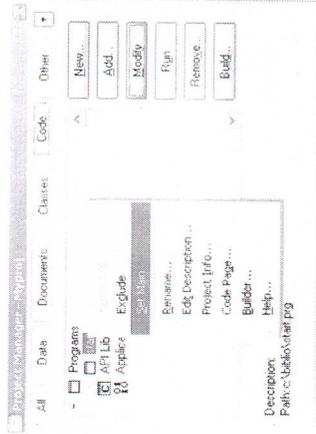
СИСТЕМНЫЕ УСТАНОВКИ

ОПРЕДЕЛЕНИЕ  
ОКНА  
ПРИЛОЖЕНИЯ

ЗАГРУЗКА МЕНЮ И  
ОЖИДАНИЕ СОБЫТИЯ

После ввода текста программы сохраните ее с именем **start**.

- Установите для программы свойство главного файла – из контекстного меню пункт **Set Main**. Убедитесь, что имя программы выделилось жирным шрифтом.



### 2. **Компильция проекта приложения.**

- **Задание:** выполнить компильцию приложения и создать запускаемый файл.
  - Для компильции приложения используется кнопка проекта **Build**. Нажмите данную кнопку. Появится диалоговое окно **Build Options**, в котором можно создать адаптированное приложение или обновить существующий проект.

#### Настройка диалогового окна **Build**

**Rebuild Project** - создает и строит файл проекта. Данная опция соответствует команде **BUILD PROJECT**.

**Application** - строит проект, компилирует устаревшие файлы и создает единый файл **APP**. Данная опция соответствует команде **BUILD APP**.

**Win32 Executable** - создает исполняемый файл на основе проекта. Данная опция соответствует команде **BUILD EXE**; она доступна только в Professional Edition of Visual FoxPro.

**Recompile All Files** - перекомпилирует все файлы, входящие в проект и создает объектный файл для каждого исходного файла.

**Display Errors** - показывает ошибки этапа компильции в окне редактирования после завершения построения.

- Выберите в данном окне пункты **Win32 Executable** и **Recompile All Files** и нажмите кнопку **OK**.

**ВНИМАНИЕ!** Если в объектах проекта имеются недоработки, то будет выведено сообщение об ошибке. Его можно временно игнорировать, нажав кнопку **Ignore**.

- Для просмотра имеющихся ошибок приложения используется пункт меню **Project – Errors**.

➤ Закомпилируйте приложение заново и запустите на выполнение. Проверьте работу созданной вами формы.

5. **Покажите результатам преподавателю.**

**Контрольные вопросы:**

1. **Что такое компиляция?**
2. **Какие этапы компиляции?**

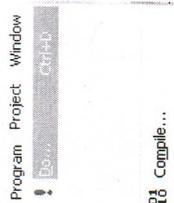


Просмотрите ошибки. В файле ошибок будет указан тип ошибки, и файл, в котором ошибка обнаружена.

**После устранения всех ошибок необходимо повторить процедуру компиляции приложения.**

3. **Запуск приложения.**

➤ Для запуска приложения используется пункт меню **Program – Do**.



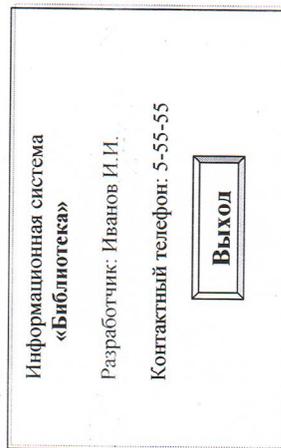
После этого откроется окно **DO**. Выберите в списке свой созданный файл с расширением **.exe** и запустите его.

**Проверьте работу приложения.**

4. **Дополнение приложения новыми опциями.**

*Задание: создать новую форму с информацией о разработчике и программе.*

➤ Создайте новую форму **info** с данными о программе и разработчике (свои данные). Установите необходимые свойства и события для формы и ее объектов.



## Практическая работа №17 «Мониторинг сетевого графика».

### Тема 1.3 Организация защиты данных в хранилищах

#### Цель работы: «Преобразование реляционной БД»

**Материально-техническое обеспечение:** Компьютер, операционная система Windows 7, FoxPro

#### **Краткие теоретические сведения:**

Средства защиты БД, реализованные в ACCESS, позволяют предотвратить умышленные или случайные просмотр, изменение и удаление информации лицами, которые не имеют соответствующих прав доступа. Эти средства особенно важны при функционировании БД в сети.

В ACCESS предусмотрены различные уровни защиты данных и администрирования доступа к ним. Возможности ACCESS позволяют обеспечить безопасность как самого приложения, так и файла БД. Простейшим средством защиты БД от несанкционированного доступа является пароль.

После того как пароль установлен, при каждом открытии БД будет появляться диалоговое окно, в которое требуется ввести пароль. Пользователи смогут открыть БД, только после ввода правильного пароля. Этот способ достаточно надежен, поскольку ACCESS шифрует пароль, так что к нему нет прямого доступа при чтении файла БД. Недостаток такого способа защиты в том, что он применяется только при открытии БД. После открытия БД все объекты становятся доступными для пользователя (если не определена защита на уровне пользователей). Для БД, которая совместно используется небольшой группой пользователей или на автономном компьютере, установка пароля обычно оказывается достаточной.

#### **Порядок выполнения практической работы:**

1. Изучить теоретический материал.
2. Выполнить предлагаемые задания.
3. Ответить на контрольные вопросы и предоставить в тетради в виде отчета. Отчет должен включать:
  - номер, наименование практической работы и тему;
  - ответы на контрольные вопросы;
  - выводы.
4. Выполненную работу и отчет по проделанной работе предъявить преподавателю.

#### **Задания для выполнения практической работы:**

1) Установите пароль на любую из созданных вами баз данных. Для этого закройте свою БД, не выходя из Access, затем снова откройте ее, щелкнув в окне открытия БД справа от кнопки Открыть и выбрав «Монопольно». Открыв БД, в меню Сервис выберите Задать пароль и следуйте дальнейшим указаниям Access.

2) При следующем открытии БД система запросит установленный вами пароль. Снять пароль можно, снова открыв БД монопольно и выбрав в меню Сервис Удалить пароль. Естественно, перед удалением пароля система потребует ввести его, чтобы никто, кроме вас, не смог произвести эту операцию. Удалите пароль со своей базы данных.

3) Усовершенствовать защиту позволяют средства *поддержки рабочих групп, ведения учетных (регистрационных) записей, задания прав владения и прав доступа*. С помощью средств защиты можно указать, какие операции по обработке объектов БД разрешается выполнять пользователю или группе пользователей. О каждом пользователе или группе ведутся учетные записи с указанием прав доступа.

Рабочей называется группа пользователей, работающих с одной БД и имеющих общий файл рабочей группы. Файл рабочей группы - это системный файл с информацией о группе

пользователей, работающих в многопользовательской среде с БД. В файлах рабочих групп хранятся учетные записи, пароли, а также данные о правах доступа к объектам БД.

При установке ACCESS по умолчанию создается стандартный файл рабочей группы SYSTEM.MDW, сохраняемый в папке, в которой установлен ACCESS. Он используется программой до появления нового файла рабочей группы. Изменение стандартного или создание нового файла рабочей группы выполняется программой Администратор рабочих групп (Wkgadm), находящаяся в подпапке System папки Windows. По умолчанию все файлы рабочей группы ACCESS имеют расширение .MDW.

Текущая рабочая группа указывается в файле рабочей группы, имя которого отображается в диалоговом окне программы «Администратор рабочих групп». Она находится среди системных файлов Windows, найти ее можно с помощью поиска в программе Проводник. Найдите и запустите Администратор рабочих групп.

Для изменения файла рабочей группы нажать кнопку Связать (Связь), вследствие чего на экране появится диалоговое окно Файл рабочей группы.

В поле База данных ввести полное имя нового файла рабочей группы. Этот файл можно найти, используя кнопку Обзор... После нажатия кнопки ОК программа сообщит об установлении связи с новой рабочей группой.

Для создания нового файла рабочей группы необходимо нажать кнопку Создать в окне Администратор рабочих групп, при этом откроется диалоговое окно Сведения о владельце рабочей группы (Имя, Организация и Код группы - любую комбинацию из букв и цифр, не превышающую 20 символов), ввести соответствующие данные. По умолчанию отображается информация о пользователе, указанном при установке Microsoft Office 97.

После нажатия ОК на экране отобразится диалоговое окно Файл рабочей группы, поле База данных которого содержит полное имя существующего файла рабочей группы. При необходимости ввести новое имя и нажать кнопку ОК. Новый системный файл рабочей группы используется только после следующего запуска ACCESS.

Аналогичным образом для каждой группы пользователей можно создать файлы рабочих групп, которые рекомендуется поместить в одну папку. Файл рабочей группы не обязательно помещать в папку с файлом БД.

Сведения об имени, организации и коде рабочей группы необходимо записать и сохранить, т.к. эти данные могут понадобиться при восстановлении системного файла рабочей группы. Если по каким-либо причинам эти данные будут утеряны, восстановить БД не удастся.

После установки ACCESS пользователь получает право доступа ко всем объектам БД (становится членом группы Admins с именем Admin). Поскольку по умолчанию пароль в этой учетной записи не указывается, его не нужно вводить для входа в систему.

Члены группы Admins (администраторы) имеют право на модификацию БД. Пока не будет задана регистрация входа в систему всех членов рабочей группы, ACCESS будет разрешать вход в систему используя предопределенную регистрационную запись Admin. Чтобы задать регистрацию входа в систему и тем самым устранить произвольный доступ к ней, следует установить пароль для администратора в регистрационной записи Admin. В противном случае при каждом запуске ACCESS администратор будет регистрироваться как пользователь Admin без указания пароля.

Чтобы регистрировался вход в систему необходимо выполнить последовательность действий:

q Запустить ACCESS, указав рабочую группу, и открыть БД.

q В подменю Защита меню Сервис активизировать команду Пользователи и группы.

q В открывшемся диалоговом окне перейти на вкладку Изменение пароля, ввести в поле Новый пароль задуманный пароль и подтвердить его в поле Подтверждение. Поле Текущий должно остаться пустым. Затем - кнопка ОК.

При вводе пароля вместо введенных символов отображаются звездочки (\*). Пароль может иметь длину от 1 до 14 знаков и включать любые символы, кроме 0 кода ASCII. Следует помнить, что при вводе пароля различаются строчные и прописные символы.

При следующем запуске ACCESS на экране появится диалоговое окно Вход для ввода пароля. В поле Имя следует ввести имя пользователя, а в поле Пароль - пароль. После нажатия кнопки ОК программа продолжит процесс запуска, если пароль введен правильно. Удалить пароль можно, выполнив последовательность действий:

- q В подменю Защита меню Сервис активизировать команду Пользователи и группы.
- q В открывшемся диалоговом окне перейти на вкладку Пользователи и, удостоверившись, что в поле Имя указано правильное имя пользователя, нажать кнопку Снять пароль.

Таким же образом устанавливается и изменяется пароль для остальных пользователей системы. Сначала они входят в систему под своим именем, не вводя пароль, затем указывают его.

4) *Учетные записи.* После создания рабочей группы можно приступить к внесению учетных (регистрционных) записей. По умолчанию создается учетная запись Admin, а также учетные записи групп Admins (Администраторы), Users (Пользователи) и предоставляются права доступа ко всем объектам.

Учетная запись администратора включена в рабочую группу Admins. Администратор имеет право доступа ко всем объектам, созданным в этой группе. Кроме администратора может быть указан владелец БД. В системе обеспечения безопасности ACCESS 97 владельцы объектов имеют особый статус. По умолчанию пользователь, создавший объект, становится владельцем объекта и прав на работу с ним.

Администраторы и владельцы наделены особыми правами:

- q Администратор БД всегда может получить права доступа ко всем объектам, созданным членами данной рабочей группы;
  - q Владелец БД всегда может открыть БД;
  - q Владелец объекта наделен полными правами доступа к этому объекту.
- q Так как регистрационная запись Admin одинакова для всех копий ACCESS, для защиты БД необходимо включить нового пользователя в группу Admins или заменить администратора Admin.

q Чтобы включить нового пользователя в группу Admins, необходимо выполнить действия:

- o Запустить ACCESS, выбрав рабочую группу и открыть любую БД
- o В подменю Защита меню Сервис активизировать команду Пользователи и группы.
- o В открывшемся диалоговом окне перейти на вкладку Пользователи и нажать кнопку Создать.

o В появившемся диалоговом окне в поле Имя ввести имя учетной записи (пользователя), а в поле Код - уникальный код записи. Затем - кнопка ОК.

o Имя пользователя может иметь длину от 1 до 20 символов и включать буквы, цифры, пробелы и другие символы за исключением символов: "\ | < > + = , ? \* и управляющих символов (с кодами ASCII от 00 до 31). Кроме того, имя записи не может начинаться с пробела.

o После возвращения в окно Пользователи и группы выбрать в списке Имеющиеся группы группу Admins и нажать кнопку Добавить. Группа Admins будет помещена в список Участие в группе, что обеспечит ввод учетной записи в группу Admins. В завершение нажать ОК.

Такая же последовательность операций выполняется при создании учетной записи владельца, которая, в зависимости от требования защиты, может быть внесена в группу Admins или в другую группу.

Кроме учетных записей администратора и владельца в ACCESS 97 создаются записи для пользователей и групп пользователей, которые могут иметь одинаковые права.

#### Примечание:

Учетные записи пользователей должны создаваться в рабочей группе, которая будет применяться при запуске приложения.

Для создания новой группы пользователей и соответствующей учетной записи следует активизировать команду Пользователи и группы подменю Защита меню Сервис, в открывшемся диалоговом окне выбрать вкладку Группы и нажать кнопку Создать.

В открывшемся диалоговом окне в поле Имя указывается имя группы, а в поле Код - уникальный код (любая комбинация символов длиной от 4 до 20 знаков).

На имя группы накладываются те же ограничения, что и на имя пользователя.

Удаление учетной записи или группы из рабочей группы осуществляется следующим образом:

q Запустить ACCESS с выбранной рабочей группой и открыть БД.

q В подменю Защита меню Сервис активизировать команду Пользователи и группы.

q В открывшемся диалоговом окне перейти на вкладку Пользователи (при удалении учетной записи пользователя) или Группы (при удалении учетной записи группы).

q В списке Имя выбрать имя пользователя или группы для удаления, нажать кнопку Удалить, затем подтвердить удаление, нажав кнопку Да.

q Закрывать окно Пользователи и группы.

После создания учетных записей пользователей и групп взаимосвязь между ними можно проследить, нажав кнопку Отчет о защите. При этом появляется диалоговое окно, в котором можно выбрать необходимые сведения и вывести их на печать.

5) *Изменение прав владельца.* После инсталляции ACCESS пользователь Admin является владельцем любой БД и всех объектов. Поскольку для него не устанавливается пароль, для защиты БД от несанкционированного доступа необходимо изменить права владельца базой и всеми объектами.

Существует несколько способов смены владельца объектов БД:

q импортирование всех объектов БД в новый файл;

q использование вкладки Смена владельца диалогового окна Разрешения.

Для первого способа последовательность действий такая:

q Запустить ACCESS с выбранной рабочей группой.

q Войти в систему под именем нового владельца БД

q Создать новую (пустую) БД

q В подменю Внешние данные меню Файл активизировать команду Импорт.

q В окне Импорт указать БД, владельца которой необходимо сменить, и скопировать все ее объекты в новую БД.

Более простой способ смены владельца БД состоит в следующем:

q Открыть БД и выбрать в меню Сервис команду Защита.

q В открывшемся подменю активизировать команду Разрешения.

q В появившемся диалоговом окне открыть вкладку Смена владельца.

q Выбрать тип объекта в списке Тип объекта. В области Объект отобразится список всех объектов выбранного типа, а в области Текущий владелец - имена владельцев.

q Выделить в списке Объекты те объекты, владельца которых необходимо сменить.

q В поле Новый владелец указать учетную запись пользователя, которому предоставляются права владельца этих объектов. Если в качестве владельца будет выступать группа, следует предварительно активизировать переключатель Группы.

q Нажать кнопку Сменить владельца.

Присвоение и удаление прав доступа. Администратор БД предоставляет всем или некоторым членам рабочей группы права доступа к различным объектам базы. Права доступа хранятся в файле БД и характеризуют ее объекты.

К разграничению прав доступа пользователей и групп можно приступить после создания рабочей группы, определения администратора и владельца БД, а также создания учетных

записей пользователей и групп. Пользователь наследует права той группы, к которой принадлежит.

Перечень прав доступа, определенных в ACCESS приведен в следующей таблице:

Право доступа	Действие	Объекты доступа
Открытие/Запуск макроса	Открытие БД, формы, отчета или запуска макроса	БД, формы, отчеты и макросы
Чтение макета	Просмотр объектов в режиме конструктора	Таблицы, запросы, формы, отчеты, макросы, модули
Изменение макета	Просмотр, изменение и удаление объектов в режиме конструктора	Таблицы, запросы, формы, отчеты, макросы и модули
Администратор	Полный доступ к объектам и данным, включая возможность присвоения прав доступа	БД, таблицы, запросы, формы, отчеты, макросы и модули
Чтение данных	Просмотр данных	Таблицы и запросы
Обновление данных	Просмотр и изменение данных без вставки и удаления	Таблицы и запросы
Вставка данных	Просмотр и вставка данных без изменения и удаления	Таблицы и запросы
Удаление данных	Просмотр и удаление данных	Таблицы и запросы
Монопольный доступ	Открытие БД в монопольном режиме	БД

Наряду с правами доступа к существующим объектам предоставляются и права доступа к новым объектам. Из таблицы видно, что некоторые права доступа обуславливают наличие других прав. Так таблица, в которой разрешено обновление данных, доступна для чтения данных и макета. При наличии прав администратора пользователю доступно все перечисленные выше права.

Определить права доступа к некоторому объекту может владелец этого объекта, администратор в рабочей группе Admins или пользователь, с правами администратора для данного объекта.

Все права доступа к объекту сохраняются при его изменении только в том случае, если не применялся буфер обмена или не выполнялся импорт/экспорт объекта. Кроме того, все связанные с объектом права доступа могут быть утеряны при сохранении объекта под новым именем посредством команды Сохранить как/экспорт.

Чтобы предоставить права доступа к объектам БД, необходимо выполнить действия:

q Открыть БД. При этом пользователь, вошедший в систему, должен иметь право доступа к объектам.

q Выбрать в меню Сервис команду Защита. В открывшемся подменю активизировать команду Разрешения.

q В появившемся диалоговом окне перейти на вкладку Разрешения.

q В поле Тип объекта выбрать тип объекта, на который распространяются новые права.

q В списке Имя объекта отметить объекты, доступ к которым будет разрешен пользователю или группе, указанным в списке Пользователи и группы.

q Выбрать право доступа к отмеченным объектам.

q Нажать ОК.

Шифрование БД. Шифрование - это защита БД от несанкционированного доступа с помощью текстового редактора или средств работы с файлами, например, входящих в состав Windows или Norton Utilities. Информация в зашифрованной БД недоступна для чтения. Шифрование несколько замедляет работу ACCESS, т.к. расходуется время на шифрование и дешифрование файлов.

Шифрование и дешифрование БД могут производиться только членами группы Admins. Эти операции выполняются следующим образом:

q Запустить ACCESS с выбранной рабочей группой.

q В подменю Защита меню Сервис активизировать команду Шифровать/Дешифровать, вследствие чего на экране появится окно выбора БД для шифрования.

q Выбрать БД, которую необходимо зашифровать или дешифровать, и нажать ОК.

q Если выбранная БД не зашифрована, откроется диалоговое окно Шифрование базы данных под именем, в котором программа предложит новое имя для зашифрованной БД. Если же выбранная БД зашифрована, появится окно Дешифрование базы данных под именем.

q Выбрать имя файла, и нажать кнопку Сохранить.