

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение высшего образования  
**«МОСКОВСКИЙ АВТОМОБИЛЬНО-ДОРОЖНЫЙ  
ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ (МАДИ)»**  
ВОЛЖСКИЙ ФИЛИАЛ



Кафедра гуманитарных и естественнонаучных дисциплин

**Методические указания к лабораторным работам  
по дисциплине  
БАЗЫ ДАННЫХ**

Направление подготовки

***09.03.01 Информатика и вычислительная техника***

Направленность (профиль, специализация) образовательной программы

***«Автоматизированные системы обработки информации и управления»***

Квалификация

бакалавр

Чебоксары  
2019

## Оглавление

|                                                                                                        |    |
|--------------------------------------------------------------------------------------------------------|----|
| Лабораторная работа №1. «Access. Создание таблиц БД и связей между ними» .....                         | 3  |
| Лабораторная работа № 2 «Отбор данных с помощью запросов».....                                         | 10 |
| Лабораторная работа №3 «Разработка экранных форм базы данных» .....                                    | 13 |
| Лабораторная работа №4 «Формирование отчетов».....                                                     | 17 |
| Лабораторная работа №5 «Разработка реляционной базы данных» .....                                      | 19 |
| Лабораторная работа № 6. «Изучение возможностей конструктора баз данных Runa» .....                    | 23 |
| Лабораторная работа № 7. «Типы связей таблиц с дочерним объектом в форме» .....                        | 31 |
| Лабораторная работа № 8. «Формирование шаблонов вывод списка записей в документ Microsoft Word » ..... | 37 |

## **Лабораторная работа №1. «Access. Создание таблиц БД и связей между ними»**

Microsoft Access – это функционально полная реляционная система управления базами данных (СУБД), в которой предусмотрены все необходимые средства для определения и обработки данных, а также для управления ими при работе с большими объемами информации. Кроме того, Access – мощная платформа разработки офисных приложений с чрезвычайно гибкой и функциональной средой пользователя.

Основные возможности Access:

- управление данными во внешней и внутренней памяти и обеспечение эффективного доступа к ним в процессе решения задач;
- поддержание целостности данных и управление транзакциями;
- обеспечение безопасности данных;
- ведение систематического журнала изменений в БД для обеспечения восстановления БД после технического или программного сбоя;
- проверка допустимости введенного значения;
- автоматизация многих процессов без программирования;
- наличие встроенного языка программирования VBA (Visual Basic for Applications), с помощью которого можно программировать сложные процедуры обработки данных;
- взаимодействие Access с другими приложениями;
- средства для работы в Internet
- инструменты для управления базами данных: конструкторы таблиц, форм, запросов, отчетов и др.

Через ряд окон интерфейса можно выполнять следующие процедуры разработки:

- добавлять, изменять и удалять объекты в приложениях, базах данных и компонентах;
- редактировать методы и свойства объектов;
- устанавливать соединения между объектами интерфейса и объектами данных;
- просматривать и выбирать компоненты текущих объектов и библиотек;
- отлаживать код программных процедур;
- тестировать объекты приложения, определять их вид и поведение во время выполнения.

*База данных (БД)* – упорядоченная совокупность данных, предназначенных для хранения, накопления и обработки с помощью ЭВМ. Для создания и ведения баз данных (их обновления, обеспечения доступа по запросам и выдачи данных по ним пользователю) используется набор языковых и программных средств, называемых *системой управления базами данных (СУБД)*.

### **Объекты базы данных Access**

К объектам базы данных Access относятся:

1. *Таблицы* – предназначены для упорядоченного хранения данных.
2. *Запросы* – предназначены для поиска, извлечения данных и выполнения вычислений.
3. *Формы* – предназначены для удобного просмотра, изменения и добавления данных в таблицах.
4. *Отчеты* – используются для анализа и печати данных.
5. *Страницы доступа к данным* – предназначены для просмотра, ввода, обновления и анализа данных через сеть или из любого места компьютера.
6. *Макросы* – используются для выполнения часто встречающегося набора макрокоманд, осуществляющих обработку данных.
7. *Модули* – предназначены для описания инструкций и процедур на языке VBA.

Основным объектом базы данных является таблица, которая состоит из записей (строк) и полей (столбцов). На пересечении записи и поля образуется ячейка, в которой содержатся данные.

Каждому полю таблицы присваивается *уникальное имя*, которое не может содержать более 64 символов. В каждом поле содержатся данные одного типа.

**Таблица 1.** Типы данных

| Тип                | Описание                                                                                                                                                                                                                                                                                                                                   |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Текстовый          | Используется для хранения символьных или числовых данных, не требующих вычислений. В свойстве <i>Размер поля</i> задается максимальное количество символов, которые могут быть введены в данное поле. По умолчанию размер устанавливается в 50 знаков. Максимальное количество символов, которые могут содержаться в текстовом поле, – 255 |
| Поле МЕМО          | Предназначено для ввода текстовой информации, по объему превышающей 255 символов; может содержать до 65 536 символов                                                                                                                                                                                                                       |
| Числовой           | Предназначен для хранения числовых данных, используемых в математических расчетах. На вкладках <i>Общие</i> и <i>Подстановка</i> можно установить свойства числового поля, среди которых <i>Размер поля</i> , <i>Формат поля</i> , <i>Число десятичных знаков</i>                                                                          |
| Дата/Время         | Используется для представления даты и времени. Выбор конкретного формата даты или времени устанавливается в свойстве <i>Формат даты</i>                                                                                                                                                                                                    |
| Денежный           | Предназначен для хранения данных, точность представления которых колеблется от 1 до 4 знаков после запятой. Целая часть может содержать до 15 десятичных знаков                                                                                                                                                                            |
| Счетчик            | Предназначен для автоматической вставки уникальных последовательных (увеличивающихся на 1) или случайных чисел в качестве номера новой записи. Номер, присвоенный записи, не может быть удален или изменен. Поля с этим типом данных используются в качестве ключевых полей таблицы                                                        |
| Логический         | Предназначен для хранения одного из двух значений, интерпретируемых как «Да / Нет», «Истина / Ложь», «Вкл. / Выкл.»                                                                                                                                                                                                                        |
| Поле объекта OLE   | Содержит данные, созданные в других программах, которые используют протокол OLE. Это могут быть, например, документы Word, электронные таблицы Excel, рисунки, звуковые и видеозаписи и др. Объекты OLE связываются с базой данных Access или внедряются в нее. Сортировать, группировать и индексировать поля объектов OLE нельзя         |
| Гиперссылка        | Специальный тип, предназначенный для хранения гиперссылок                                                                                                                                                                                                                                                                                  |
| Мастер подстановок | Предназначен для автоматического определения поля. С его помощью будет создано поле со списком, из которого можно выбирать данные, содержащиеся в другой таблице или в наборе постоянных значений                                                                                                                                          |

Запустите Microsoft Access 2007.

На вкладке ленты *Создание* в панели инструментов *Таблицы* нажмите на кнопку *Конструктор таблиц*. Введите имена полей и укажите типы данных, к которым они относятся (таблица 2).

Таблица 2. Имена полей и типы данных

| Имя поля         | Тип данных       |
|------------------|------------------|
| Код сотрудника   | Счетчик          |
| Фамилия          | Текстовый        |
| Имя              | Текстовый        |
| Отчество         | Текстовый        |
| Должность        | Текстовый        |
| Адрес            | Текстовый        |
| Дата рождения    | Дата/Время       |
| Телефон          | Текстовый        |
| Заработная плата | Денежный         |
| Фото             | Поле объекта OLE |
| Эл_почта         | Гиперссылка      |

Выйдите из режима Конструктора, предварительно сохранив таблицу под именем «Сотрудники»; ключевые поля не задавайте (рис.1).

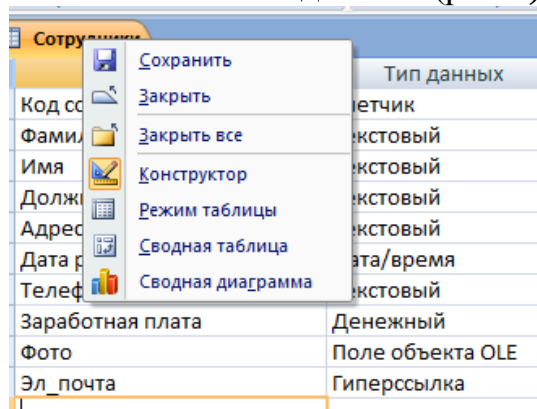


Рис. 1

Откройте таблицу «Сотрудники» двойным щелчком мыши и заполните в ней 15 строк.

Добавьте поле «Отчество», для этого:

- установите курсор на поле, перед которым нужно вставить новый столбец;
- выполните команду: вкладка ленты *Режим таблицы* → панель инструментов *Поля и столбцы* → *Вставить*;
- щелкнув два раза на *Поле1*, переименуйте его в «Отчество».

Перейдите в режим Конструктора командой: вкладка ленты *Главная* → *Вид* → *Конструктор*.

10. Для поля «Дата рождения» установите тип данных *Дата / время*; в свойствах поля выберите *Краткий формат даты* (Рис. 2).



заполняется поле «Семейное положение». Чтобы фамилия была постоянно видна при заполнении таблицы, необходимо воспользоваться командой «Закрепить столбцы» из контекстного меню поля «Фамилия» (рис.3).

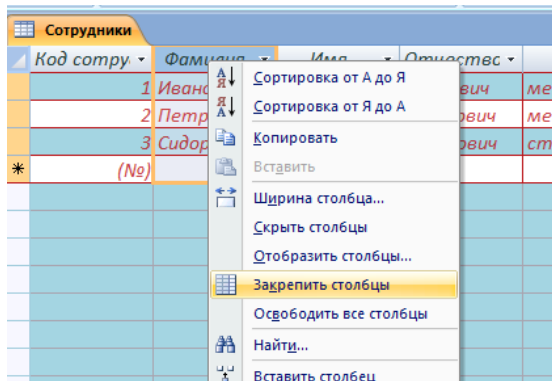


Рис. 3

### Создание связей между таблицами

Создадим базу данных «Предприятие». Сотрудники организации работают с клиентами и выполняют их заказы.

Необходимо создать 3 таблицы: *Сотрудники*, *Клиенты* и *Заказы*. Таблицу *Сотрудники* создали в пункте 1.

#### Клиенты

| Имя поля          | Тип данных  |
|-------------------|-------------|
| Код клиента       | Счетчик     |
| Название<br>фирмы | Текстовый   |
| Адрес             | Текстовый   |
| Телефон           | Текстовый   |
| Факс              | Числовой    |
| Эл_почта          | Гиперссылка |
| Разное            | Поле МЕМО   |

#### Заказы

| Имя поля              | Тип данных |
|-----------------------|------------|
| Код заказа            | Счетчик    |
| Код клиента           | Числовой   |
| Код сотрудника        | Числовой   |
| Дата размещения       | Дата/Время |
| Дата исполнения       | Дата/Время |
| Сумма                 | Денежный   |
| Отметка<br>выполнении | Логический |

Отдельные таблицы, содержащие информацию, необходимо связать в единую структуру базы данных. Для связывания таблиц следует задать ключевые поля. Ключ состоит из одного или нескольких полей, значения которых однозначно определяют каждую запись в таблице. Наиболее подходящим в качестве ключевого поля является «Счетчик», так как значения в данном поле являются уникальными (т. е. исключают повторы)

Откройте таблицу *Сотрудники* в режиме Конструктора. Нажмите правой кнопкой мыши на поле «Код сотрудника» и в появившемся контекстном меню

выберите команду «*Ключевое поле*». Если в таблице необходимо установить несколько ключевых полей, то выделить их можно, удерживая клавишу *Ctrl*.

Для таблицы «*Клиенты*» установите ключевое поле «*Код клиента*», а для таблицы «*Заказы*» – «*Код заказа*».

Таблица «*Заказы*» содержит поля «*Код сотрудника*» и «*Код клиента*». При их заполнении могут возникнуть некоторые трудности, так как не всегда удастся запомнить все предприятия, с которыми работает фирма, и всех сотрудников с номером кода. Для удобства можно создать раскрывающиеся списки с помощью «*Мастера подстановок*».

Откройте таблицу *Заказы* в режиме Конструктора.

Для поля *Код сотрудника* выберите тип данных *Мастер подстановок*.

В появившемся окне выберите команду «Объект «*столбец подстановки*» будет использовать значения из таблицы или запроса» и щелкните на кнопке «*Далее*».

В списке таблиц выберите таблицу *Сотрудники* и щелкните на кнопке «*Далее*».

В списке «*Доступные поля*» выберите поле «*Код сотрудника*» и щелкните на кнопке со стрелкой, чтобы ввести поле в список «*Выбранные поля*». Таким же образом добавьте поля *Фамилия* и *Имя* и щелкните на кнопке «*Далее*».

Выберите порядок сортировки списка по полю *Фамилия*.

В следующем диалоговом окне задайте необходимую ширину столбцов раскрывающегося списка.

Установите флажок «*Скрыть ключевой столбец*» и нажмите кнопку «*Далее*».

На последнем шаге *Мастера подстановок* замените при необходимости надпись для поля подстановок и щелкните на кнопке «*Готово*».

Аналогичным образом создайте раскрывающийся список для поля «*Код клиента*».

После создания ключевых полей можно приступить к созданию связей.

Закройте все открытые таблицы, так как создавать или изменять связи между открытыми таблицами нельзя.

Выполните команду: вкладка ленты *Работа с базами данных* → *Схема данных*.

Если ранее никаких связей между таблицами базы не было, то при открытии окна «*Схема данных*» одновременно открывается окно «*Добавление таблицы*», в котором выберите таблицы *Сотрудники*, *Клиенты* и *Заказы*.

Если связи между таблицами уже были заданы, то для добавления в схему данных новой таблицы щелкните правой кнопкой мыши на схеме данных и в контекстном меню выберите пункт «*Добавить таблицу*».

Установите связь между таблицами *Сотрудники* и *Заказы*, для этого выберите поле *Код сотрудника* в таблице *Сотрудники* и перенесите его на соответствующее поле в таблице *Заказы*.

После перетаскивания откроется диалоговое окно «*Изменение связей*» (рис. 5), в котором включите флажок «*Обеспечение условия целостности*».



Рис. 5

Это позволит предотвратить случаи удаления записей из одной таблицы, при которых связанные с ними данные других таблиц останутся без связи.

Флажки *Каскадное обновление связанных полей* и *Каскадное удаление связанных записей* обеспечивают одновременное обновление или удаление данных во всех подчиненных таблицах при их изменении в главной таблице.

Параметры связи можно изменить, нажав на кнопку *Объединение*.

После установления всех необходимых параметров нажмите кнопку *ОК*.

Связь между таблицами *Клиенты* и *Заказы* установите самостоятельно.

В результате должна получиться схема данных (рис. 6).

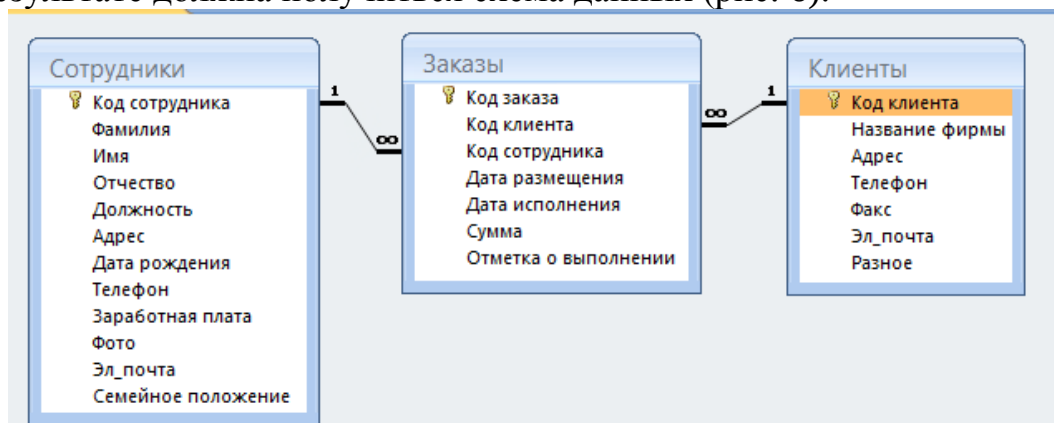


Рис. 6

В приведенном примере используются связи «один-ко-многим». На схеме данных они отображаются в виде соединительных линий со специальными значками около таблиц. Связь «один-ко-многим» помечается «1» вблизи главной таблицы (имеющей первичный ключ) и «∞» вблизи подчиненной таблицы (имеющей внешний ключ).

В таблицу *Клиенты* внесите данные о 15 предприятиях.

В таблице *Заказы* оформите 10 заявок, поступивших на предприятие.

## Лабораторная работа № 2 «Отбор данных с помощью запросов»

*Запросы* являются основным средством просмотра, отбора, изменения и анализа информации, которая содержится в одной или нескольких таблицах базы данных.

Существуют различные виды запросов, но наиболее распространенными являются *запросы на выборку*.

Откройте базу данных «*Предприятие*».

Выполните команду: вкладка ленты *Создание*→*Мастер запросов*→*Простой запрос*. В появившемся диалоговом окне укажите таблицу «*Сотрудники*» и выберите поля «*Фамилия*», «*Имя*», «*Телефон*». Нажмите кнопку «*Далее*» (рис. 7).

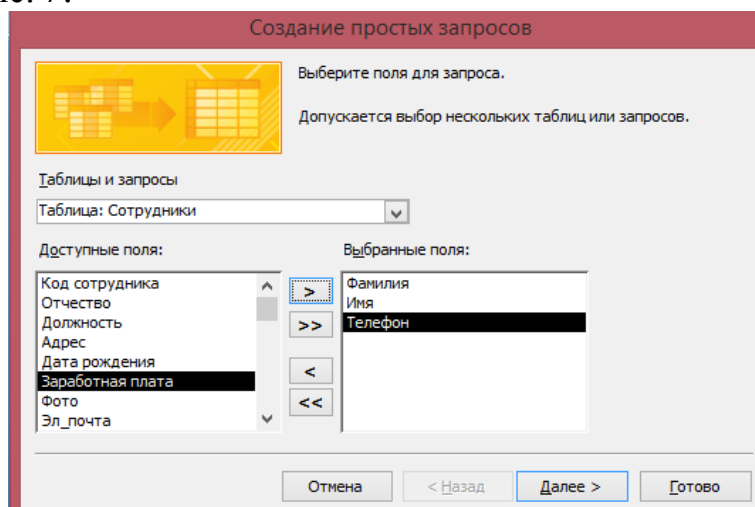


Рис. 7

Введите имя запроса – *Телефоны* – и нажмите кнопку *Готово*. Перед вами появится запрос, в котором можно просмотреть телефоны сотрудников.

Следующий запрос попробуйте создать с помощью Конструктора, для этого выполните команду: вкладка ленты *Создание* → *Конструктор запросов*.

В диалоговом окне *Добавление таблиц* выберите таблицу *Клиенты* и щелкните на кнопке *Добавить*, а затем – на кнопке *Заккрыть*.

Чтобы перенести нужные поля в бланк запроса, необходимо по ним дважды щелкнуть левой кнопкой мыши.

Чтобы отсортировать записи в поле *Название компании* в алфавитном порядке, необходимо в раскрывающемся списке строки *Сортировка* выбрать пункт «*по возрастанию*».

Сохраните запрос с именем «*Адреса клиентов*». (Рис. 8).

Создайте запрос «*Дни рождения*», в котором можно будет просмотреть дни рождения сотрудников.

Допустим, мы хотим узнать, у кого из сотрудников день рождения в январе. Для этого откройте запрос в режиме Конструктора.

В строке *Условие отбора* для поля «*Дата рождения*» введите значение *\*.01.\**. В данной записи \* означают, что дата и год рождения могут быть любыми, а месяц 1-м (т. е. январь) (рис. 9).

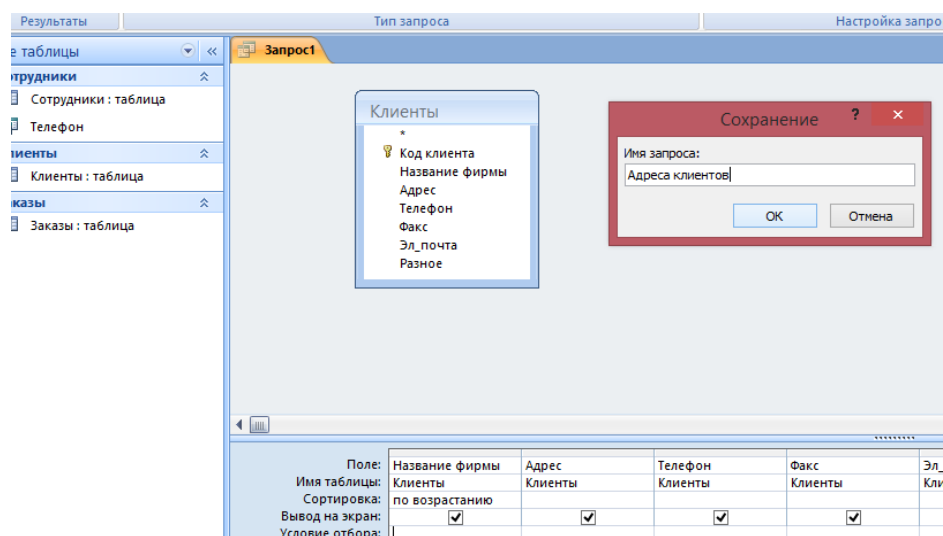


Рис.8

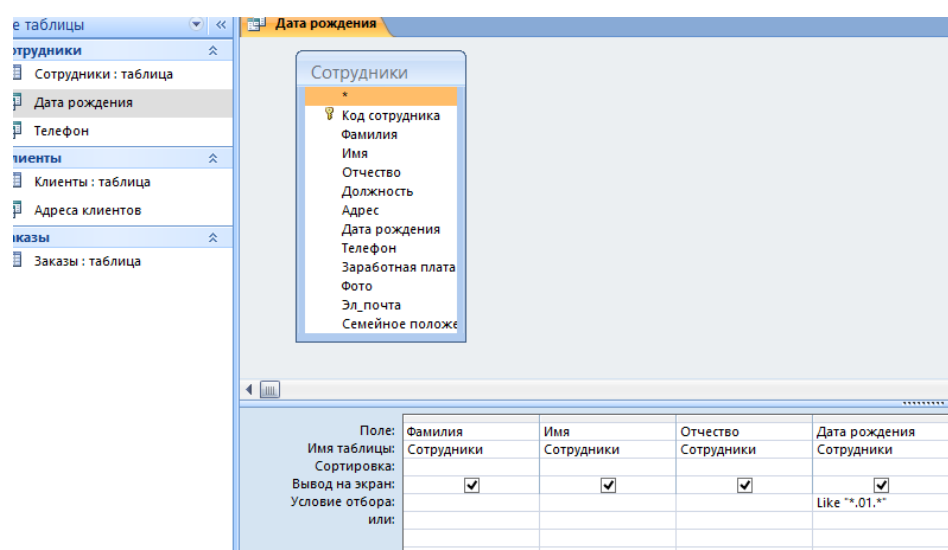


Рис.9

Закройте Конструктор и просмотрите полученный результат.

Если нужно узнать, кто родился в феврале, то придется создать новый запрос или изменить условие в существующем запросе *Дни рождения*.

Если приходится часто выполнять запрос, но каждый раз с новыми значениями условий используют *запрос с параметром*. При запуске такого запроса на экран выводится диалоговое окно для ввода значения в качестве условия отбора. Чтобы создать запрос с параметром, пользователю необходимо ввести текст сообщения в строке *Условие отбора* бланка запроса (рис. 10).

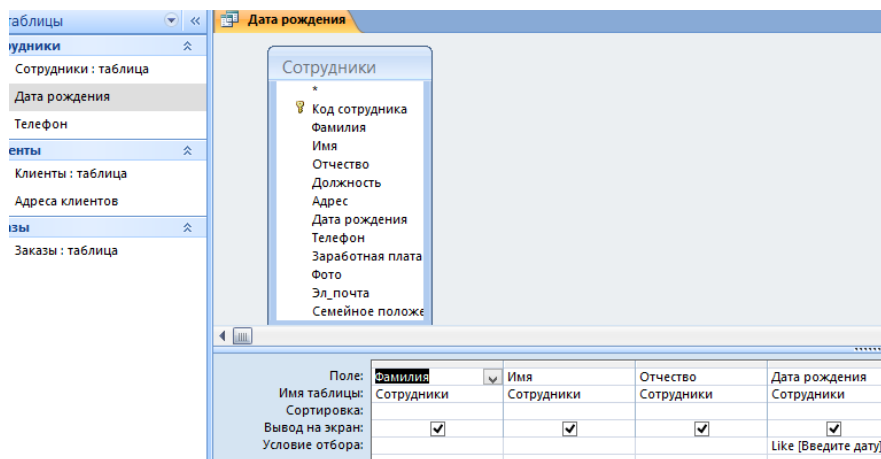


Рис. 10

Запись *Like[Введите дату]* означает, что при открытии запроса появится диалоговое окно с текстом «Введите дату» и полем для ввода условия отбора.

Измените запрос «Телефоны» так, чтобы при его запуске выводилось диалоговое окно с сообщением «Введите фамилию». Поскольку в запросе нужно вывести конкретную фамилию, в условии отбора слово *Like* писать не надо.

Измените запрос «Телефоны» так, чтобы при его запуске запрашивались не только фамилия, но и имя сотрудника.

Создайте запрос «Выполненные заказы», содержащий следующие сведения: фамилия и имя сотрудника, название фирмы, с которой он работает, отметка о выполнении и сумма заказа. Данные берутся из нескольких таблиц.

В условии отбора для логического поля *Отметка о выполнении* введите *Да*, чтобы в запросе отображались только выполненные заказы.

Сделайте так, чтобы столбец *Отметка о выполнении* не выводился на экран.

Создайте запрос *Сумма заказа*, в котором будут отображаться заказы на сумму более 50 000 руб. Измените запрос, чтобы сумма заказа была от 20 000 до 50 000 руб.

Для данных запросов в условии отбора можно использовать операторы сравнения *>*, *<*, *=*, *>=*, *<=*, *< >* и логические операторы *And*, *Or*, *Not* и др.

Иногда в запросах требуется произвести некоторые вычисления, например посчитать подоходный налог 13 % для каждой сделки. Для этого откройте запрос *Сумма заказа* в режиме Конструктора.

В пустом столбце бланка запроса щелкните правой кнопкой мыши на ячейке *Поле* и в появившемся контекстном меню выберите команду *Построить*. Перед вами появится окно *Построитель выражений*, который состоит из трех областей: поля выражения, кнопок операторов и элементов выражения. Сверху располагается поле выражения, в котором оно и создается.

Вводимые в это поле элементы выбираются в двух других областях окна Построителя.

В левом списке откройте папку *Запросы* и выделите запрос *Сумма заказа*. В среднем списке выделите поле *Сумма* и нажмите кнопку *Вставить*.

В поле выражения Построителя введите  $*0,13$ . Таким образом, посчитаем подоходный налог 13 % (рис.11)

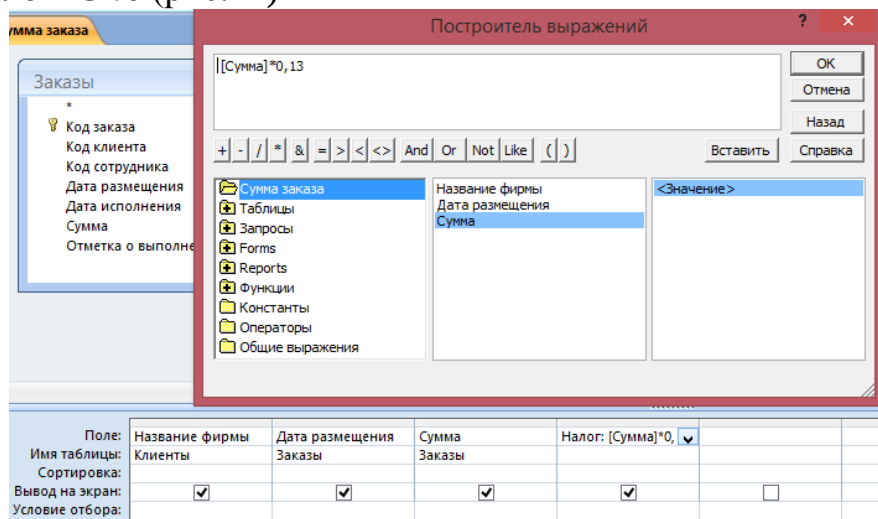


Рис. 11

Нажмите кнопку *OK*, после чего в ячейке свойства *Поле* появится значение «*Выражение1: [Сумма]\*0,13*». Замените *Выражение1* на *Налог* и закройте Конструктор. Откройте запрос и посмотрите, что у вас получилось.

Используя *Построитель выражений*, добавьте в запрос *Сумма заказа* поле *Прибыль*, в котором будет вычисляться доход от заказа (т. е. сумма минус налог).

Создайте запрос *Менеджеры*, с помощью которого в таблице *Сотрудники* найдите всех менеджеров фирмы.

### Лабораторная работа №3 «Разработка экранных форм базы данных»

*Формы* – это объекты базы данных, предназначенные для просмотра данных из таблиц и запросов, для ввода данных в базу, корректирования существующих данных и выполнения заданных действий. Форма может содержать графики, рисунки и другие внедренные объекты.

Можно вносить данные в таблицы и без помощи каких-либо форм. Но существует несколько причин, которые делают формы незаменимым средством ввода данных в базу:

- при работе с формами ограничен доступ к таблицам (самому ценному в базе данных);
- разные люди могут иметь разные права доступа к информации, хранящейся в базе. Для ввода данных им предоставляются разные формы, хотя данные из форм могут поступать в одну таблицу;
- вводить данные в форму легче, чем в таблицу, и удобнее, так как в окне формы отображается, как правило, одна запись таблицы;
- в большинстве случаев информация для баз данных берется из бумажных бланков (анкет, счетов, накладных, справок и т. д.). Экранные формы можно сделать точной копией бумажных бланков, благодаря этому

уменьшается количество ошибок при вводе и снижается утомляемость персонала.

Создавать формы можно на основе нескольких таблиц или запросов с помощью Мастера, используя средство автоформы, «вручную» в режиме Конструктора, сохраняя таблицу или запрос как форму. Созданную любым способом форму можно затем изменять в режиме Конструктора. Рассмотрим некоторые из перечисленных способов.

С помощью Мастера создайте формы *Сотрудники*, *Клиенты*, *Заказы*.

Выполните команду: вкладка ленты *Создание* → панель инструментов *Формы* → *Другие формы* → *Мастер форм*.

В диалоговом окне *Создание форм* выберите таблицы (запросы) и поля, которые будут помещены в форму. Щелкните по кнопке *Далее*.

В следующих диалоговых окнах мастера выберите внешний вид формы, стиль, задайте имя формы. Щелкните по кнопке *Готово*.

Откройте форму *Сотрудники* в режиме Конструктора. Этот режим предназначен для создания и редактирования форм.

Разместите элементы в удобном для вас порядке, измените размер и цвет текста.

В заголовок формы добавьте текст *Сотрудники фирмы*.

В примечание формы добавьте объект *Кнопка* (вкладка ленты *Конструктор* → панель инструментов *Элементы управления*).

После того как «нарисуете» кнопку указателем, на экране появится диалоговое окно *Создание кнопок* (рис. 12).

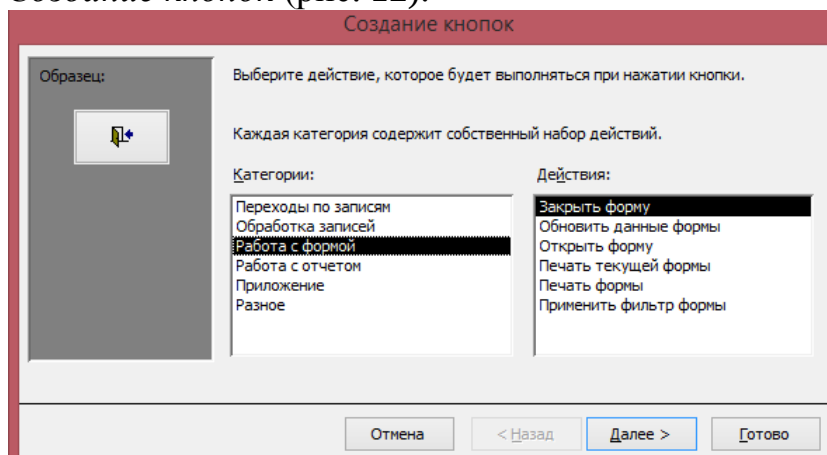


Рис. 12

В категории *Работа с формой* выберите действие *Закрытие формы* и нажмите кнопку *Далее*.

Выберите рисунок или текст, который будет размещаться на кнопке. В последнем диалоговом окне *Мастера кнопок* задайте имя кнопки и нажмите *Готово*.

Создайте кнопки *Выход из приложения*, *Поиск записи*, *Удаление записи*.

Иногда на форме требуется разместить несколько страниц, содержащих данные из различных источников, справочную или вспомогательную информацию. Для этой цели можно использовать набор вкладок.

Для создания кнопочной формы необходимо на вкладке ленты *Работа с базами данных* выбрать команду *Диспетчер кнопочных форм*. Если кнопочной формы в базе данных нет, то будет выведен запрос на подтверждение ее создания. Нажмите «Да» в диалоговом окне подтверждения.

Перед вами появится «Диспетчер кнопочных форм», в котором щелкните по кнопке «Создать». В диалоговом окне *Создание* введите имя новой кнопочной формы и нажмите *ОК* (рис. 13).

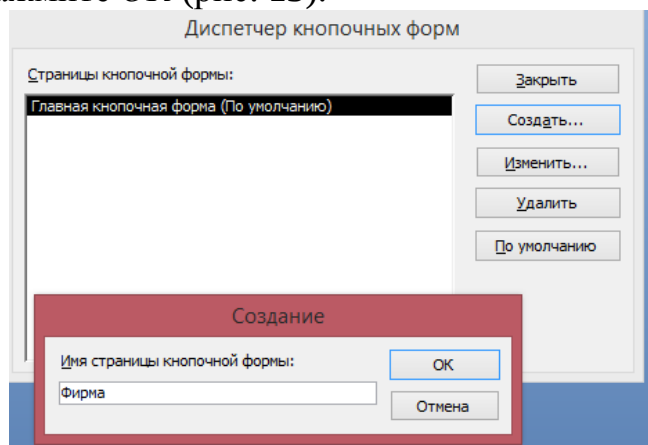


Рис. 13

Имя новой кнопочной формы добавится в список *Страницы кнопочной формы* окна *Диспетчер кнопочных форм*. Выделите имя новой кнопочной формы и щелкните по кнопке «Изменить» (рис. 14).

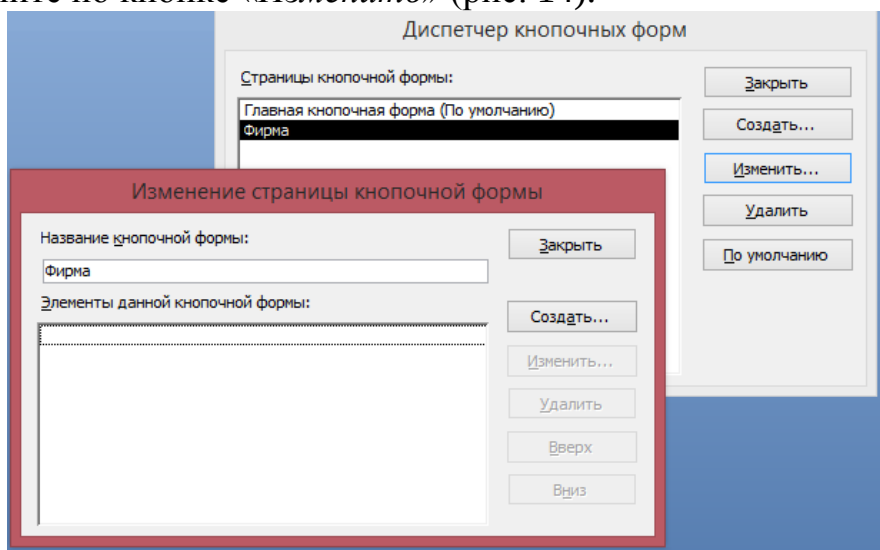


Рис. 14

В диалоговом окне «Изменение страницы кнопочной формы» щелкните по кнопке «Создать». Появится диалоговое окно «Изменение элемента кнопочной формы»

В поле «Текст» введите текст подписи для первой кнопки кнопочной формы, а затем выберите команду из раскрывающегося списка в поле «Команда».

В поле *Форма* выберите форму, для которой будет выполняться данная команда (рис. 15).



Рис. 15

Аналогичным образом добавьте кнопки *Клиенты*, *Заказы*, *Выход*.

В диалоговом окне «*Диспетчер кнопочных форм*» выберите имя вашей кнопочной формы и щелкните по кнопке «*По умолчанию*».

Рядом с названием кнопочной формы появится надпись «(по умолчанию)».

Чтобы закончить создание кнопочной формы, щелкните по кнопке *Заккрыть*. В результате должна получиться форма, представленная на рис. 16.

Рис. 16

Для того чтобы главная кнопочная форма появлялась на экране при запуске приложения, необходимо в главном меню нажать на кнопку *Параметры Access*). Для текущей базы данных установите форму просмотра – «кнопочная форма» (рис. 17)

Рис. 17



## Лабораторная работа №4 «Формирование отчетов»

Отчеты предназначены для вывода информации на печать. Часто данные в них располагаются в табличной форме. В отличие от распечаток таблиц или запросов отчет дает более широкие возможности сортировки и группировки данных, он предоставляет возможность добавлять итоговые значения, а также поясняющие надписи, колонтитулы, номера страниц, стили и различные графические элементы.

Создавать отчеты в базе данных Access можно несколькими способами:

- 1) с помощью Мастера отчетов;
- 2) на основе таблиц или запросов;
- 3) в режиме Конструктора.

В окне базы данных выполните команду: вкладка ленты *Создание* → панель инструментов *Отчеты* → *Мастер отчетов*.

Выберите из списка таблицу (или запрос), которая будет использована как источник данных (например, запрос *Адреса клиентов*).

В появившемся диалоговом окне *Создание отчетов* переместите все доступные поля в область «выбранные поля» (рис. 18).

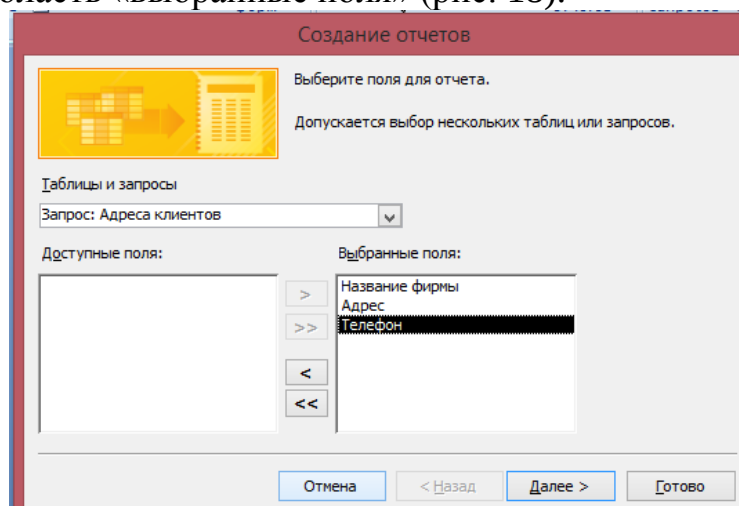


Рис. 18

С помощью «*Мастера отчетов*» создайте отчет «*Дни рождения*». В качестве источника данных используйте таблицу «*Сотрудники*».

Если требуется напечатать почтовые наклейки, Access предоставляет такую возможность. Для этого выделите таблицу *Клиенты* и выполните команду: вкладка ленты *Создание* → панель инструментов *Отчеты* → *Наклейки*.

6. В появившемся диалоговом окне укажите размер наклейки, систему единиц, тип наклейки и нажмите кнопку «*Далее*» (рис. 19).

На следующем шаге создания отчета установите шрифт, размер, цвет текста и начертание. Нажмите кнопку «*Далее*».

Выберите поля, которые будут размещаться на наклейке. Например, *Название компании*, *Адрес*, *Телефон* и *Факс*. Если на каждой наклейке требуется вывести определенный текст, то введите его в прототип наклейки.

При необходимости измените название отчета с наклейками и нажмите кнопку *Готово*.

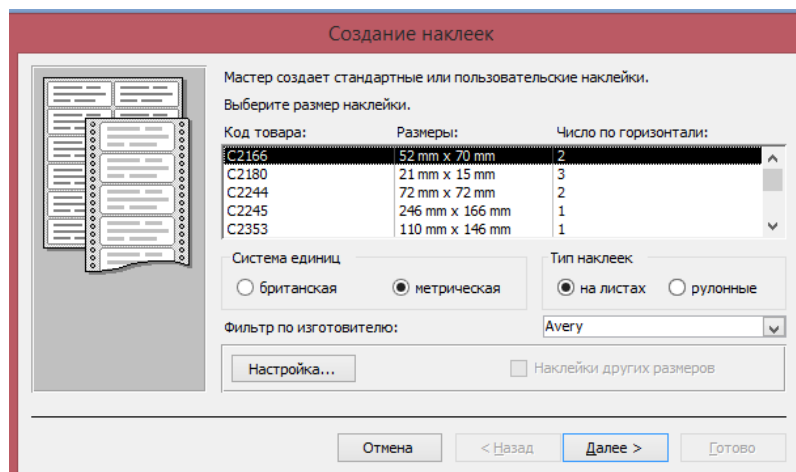


Рис. 19

Иногда в отчетах требуется вычислять итоговые значения, среднее, минимальное или максимальное значения, а также проценты. Для этого запустите *Мастер отчетов* и в качестве источника данных укажите запрос *Сумма заказа*.

В диалоговом окне *Мастера*, в котором задается порядок сортировки записей, нажмите кнопку *Итоги*.

В диалоговом окне *Итоги* для полей *Сумма* и *Налог* установите флажки в столбце *sum*, чтобы посчитать итоговую сумму (рис. 20).

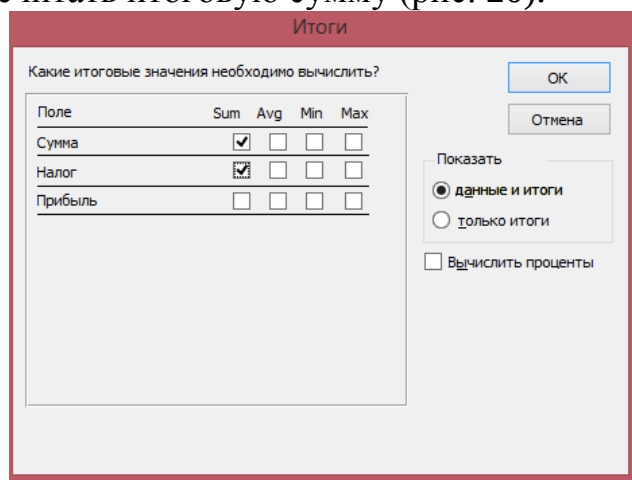


Рис. 20

Далее выполните все шаги Мастера и нажмите кнопку *Готово*.

Создайте отчет *Дни рождения*, используя в качестве источника данных запрос *Дни рождения*.

Составьте отчет *Выполненные заказы*, в котором будут данные о фирме и сумме заказа. Вычислите итоговую сумму, среднее значение (Avg) и максимальную сумму для каждой фирмы.

## Лабораторная работа №5 «Разработка реляционной базы данных»

Согласно номеру индивидуального задания (последняя цифра в зачетке) разработать реляционную базу данных.

### Вариант 1

1. Разработайте базу данных «*Электронная библиотека*», состоящую из трех таблиц со следующей структурой:

*Книги* – шифр книги (ключевое поле), автор, название, год издания, количество экземпляров.

*Читатели* – читательский билет (ключевое поле), фамилия, имя, отчество, адрес.

*Выданные книги* – шифр книги, читательский билет, дата выдачи, дата возвращения, дата фактического возвращения.

2. Установите связи между таблицами.

3. С помощью запроса отберите все книги, выпущенные с 2000 по 2015 годы.

4. Создайте запрос с параметром для отбора книг определенного автора.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### Вариант 2

1. Разработайте базу данных «*Продуктовый магазин*», которая состоит из четырех таблиц со следующей структурой:

*Товары* – код товара (ключевое поле), наименование товара, количество товара.

*Поступление товаров* – код товара, дата поступления, цена приобретения товара за единицу, код поставщика.

*Продажа товаров* – код товара, месяц продажи, проданное количество за месяц, цена продажи товара.

*Поставщики* – код поставщика (ключевое поле), название поставщика, адрес поставщика, телефон поставщика.

2. Установите связи между таблицами.

34

3. С помощью запроса отберите товары, цены которых от 100 до 450 руб.

4. Создайте запрос с параметром для отбора товаров, проданных в определенном месяце.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### Вариант 3

1. Разработайте базу данных «*Сессия*», состоящую из четырех таблиц со следующей структурой:

*Студенты* – шифр студента (ключевое поле), фамилия, имя, отчество, курс, группа.

*Экзамены* – шифр студента, дата, шифр дисциплины, оценка.

*Зачеты* – шифр студента, дата, шифр дисциплины, зачет.

*Дисциплины* – шифр дисциплины (ключевое поле), название дисциплины, количество часов.

2. Установите связи между таблицами.

3. С помощью запроса отберите студентов, сдавших экзамен на 4 или 5.

4. Создайте запрос с параметром для отбора студентов, получивших или не получивших зачет.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

#### **Вариант 4**

1. Разработайте базу данных «*Оптовый склад*», состоящую из четырех таблиц со следующей структурой:

*Склад* – код товара, количество, дата поступления.

*Товары* – код товара (ключевое поле), название товара, срок хранения.

*Заявки* – код заявки (ключевое поле), название организации, код товара, требуемое количество.

*Отпуск товаров* – код заявки (ключевое поле), код товара, отпущенное количество, дата отпуска товара.

2. Установите связи между таблицами.

3. С помощью запроса отберите товары, количество которых от 50 до 200 штук.

4. Создайте запрос с параметром для отбора товаров, поступивших на склад какого-либо числа.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

#### **Вариант 5**

1. Разработайте базу данных «*Абитуриенты*», состоящую из четырех таблиц со следующей структурой:

*Анкета* – номер абитуриента (ключевое поле), фамилия, имя, отчество, дата рождения, оконченное среднее учебное заведение (название, номер, населенный пункт), дата окончания учебного заведения, наличие красного диплома или золотой/серебряной медали, адрес, телефон, шифр специальности.

*Специальности* – шифр специальности (ключевое поле), название специальности.

*Дисциплины* – шифр дисциплины (ключевое поле), название дисциплины.

*Вступительные экзамены* – номер абитуриента, шифр дисциплины, экзаменационная оценка.

2. Установите связи между таблицами.

3. Составьте запрос для отбора студентов, сдавших экзамены без троек.

4. Создайте запрос с параметром для отбора студентов, поступающих на определенную специальность.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

#### **Вариант 6**

1. Разработайте базу данных «*Транспортные перевозки*», состоящую из трех таблиц со следующей структурой:

*Транспорт* – марка автомобиля, государственный номер (ключевое поле), расход топлива.

*Заявки* – код заявки (ключевое поле), дата заявки, название груза, количество груза, пункт отправления, пункт назначения.

*Доставка* – № п/п, дата и время отправления, дата и время прибытия, код заявки, государственный номер автомобиля, пройденное расстояние.

2. Установите связи между таблицами.

3. С помощью запроса отберите заявки с количеством груза от 100 до 500 кг.

4. Создайте запрос с параметром для отбора транспорта по марке автомобиля.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### **Вариант № 7**

1. Разработайте базу данных «*Прокат спортивного оборудования*», состоящую из трех таблиц со следующей структурой:

*Клиенты* – код клиента (ключевое поле), фамилия, имя, отчество, телефон, адрес, паспортные данные, залог.

*Склад* – код оборудования (ключевое поле), название, количество, залоговая стоимость, остаток.

*Прокат* – № п/п, клиент, оборудование, дата выдачи, срок возврата, отметка о возврате, оплата проката.

2. Установите связи между таблицами.

3. Создайте запрос для отбора оборудования с залоговой стоимостью от 10000 до 50000 руб.

4. Создайте запрос с параметром для отбора клиентов, возвративших оборудование.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### **Вариант 8**

1. Разработайте базу данных «*Банк*», состоящую из трех таблиц со следующей структурой:

*Клиенты* – код клиента (ключевое поле), фамилия, имя, отчество, паспорт, телефон, адрес, заработная плата.

*Виды кредитов* – код кредита (ключевое поле), название кредита, процентная ставка, условия предоставления.

*Предоставленные кредиты* – № п/п, клиент, кредит, дата предоставления, срок, дата возврата, сумма, отметка о возврате.

2. Установите связи между таблицами.

3. Создайте запрос для отбора клиентов, взявших кредит от 500 000 до 1000000 руб.

4. Создайте запрос с параметром для отбора кредитов по процентной ставке.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

#### **Вариант 9**

1. Разработайте базу данных «*Туристическая фирма*», состоящую из четырех таблиц со следующей структурой:

*Клиенты* – код клиента (ключевое поле), фамилия, имя, отчество, телефон, адрес, паспорт.

*Сотрудники* – код сотрудника (ключевое поле), фамилия, имя, отчество, должность, телефон, адрес, паспортные данные.

*Туристические маршруты* – код маршрута (ключевое поле), название, описание маршрута, страна, стоимость путевки, количество дней, вид транспорта.

*«Заказы»* – код заказа (ключевое поле), клиент, маршрут, сотрудник (менеджер, оформивший заказ), дата, отметка об оплате.

2. Установите связи между таблицами.

3. Создайте запрос для отбора маршрутов со стоимостью от 10000 до 20000 руб.

4. Создайте запрос с параметром для отбора клиентов, выбравших определенный вид маршрута.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

#### **Вариант 10**

1. Разработайте базу данных «*Поликлиника*», состоящую из четырех таблиц со следующей структурой:

*Врачи* – код врача (ключевое поле), ФИО, должность, специализация, стаж работы, адрес, телефон.

*Болезни* – № п/п (ключевое поле), название заболевания, рекомендации по лечению, меры профилактики.

*Пациенты* – код пациента (ключевое поле), ФИО, адрес, телефон, страховой полис, паспорт.

*Диагноз* – № п/п (ключевое поле), пациент, заболевание, лечащий врач, дата обращения, дата выздоровления.

2. Установите связи между таблицами.

3. С помощью запроса отберите врачей-стоматологов и ортопедов.

4. Создайте запрос с параметром для отбора пациентов с определенным видом заболевания.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

## Лабораторная работа № 6. «Изучение возможностей конструктора баз данных Runa»

**Цель работы:** Приобрести навыки и умения при работе с таблицами: создавать таблицы с помощью конструктора Runa, задавать тип данных, использовать свойства поля.

Конструктор баз данных Руна - простой и удобный инструмент для создания баз данных и программ учёта.

Возможности конструктора:

- создание простой структуры данных программы, аналогично построению сайтов;
- сортировка и фильтрация по любому полю;
- шаблоны документов в MS Office Word/Excel, а также OpenOffice/LibreOffice Writer/Calc;
- уровни доступа к информации различным пользователям;
- синхронизация с другими источниками данных;
- импорт и экспорт данных;
- прикрепление к данным файлов любого типа;
- работа с картографическими сервисами;
- и многое другое.

Для создания первой базы данных переходим в управление «Проекты» и вводим имя «Первая база» (рис. 6.1) и сохраняем.

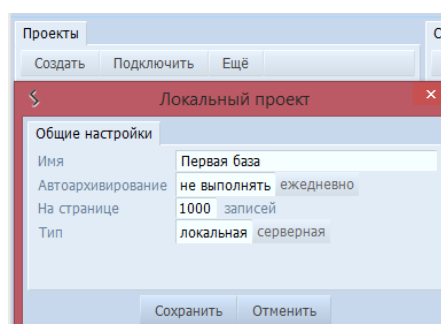


Рис. 6.1. Управление «Проекты»

В меню «Проекты» появился пункт с нашей базой, выбираем его. Так как база пустая, то создадим ее структуру (рис 6.2).

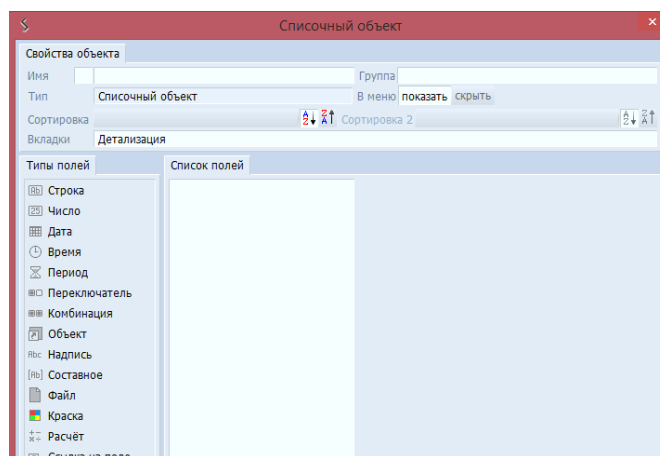


Рис. 6.2. Структура базы

Первым объектом будут клиенты.

В строке «Имя» вводим «Клиенты». Далее выбираем «Типы полей»-«Строка» и вводим имя «Наименование». По наименованию можно будет искать клиента, поэтому включаем использование фильтра: в строке «В фильтре» выбираем «Применить».

Так же в свойствах объекта включаем сортировку по полю «Наименование». Для этого в строке «Сортировка» выбираем «Наименование». Сортировку проводим по возрастанию (рис. 6.3).

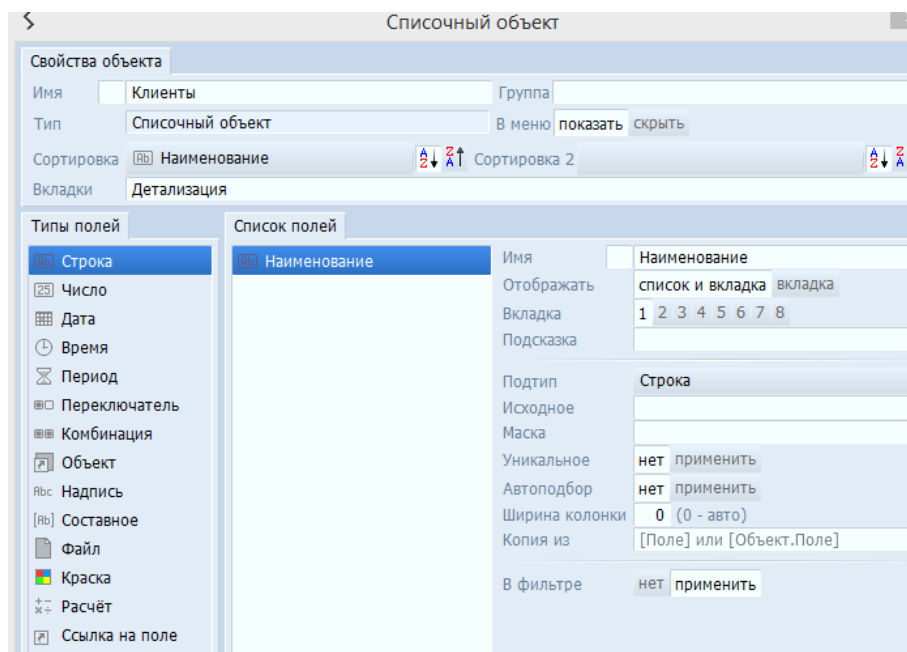


Рис. 6.3. Свойства объекта

Добавим новые поля «Телефоны» (тип поля – строка), «Дата рождения» (тип поля - дата).

Так как дата рождения может быть неизвестна, то установим ее вид как «Строка», что позволяет оставлять это поле пустым (рис 6.4).

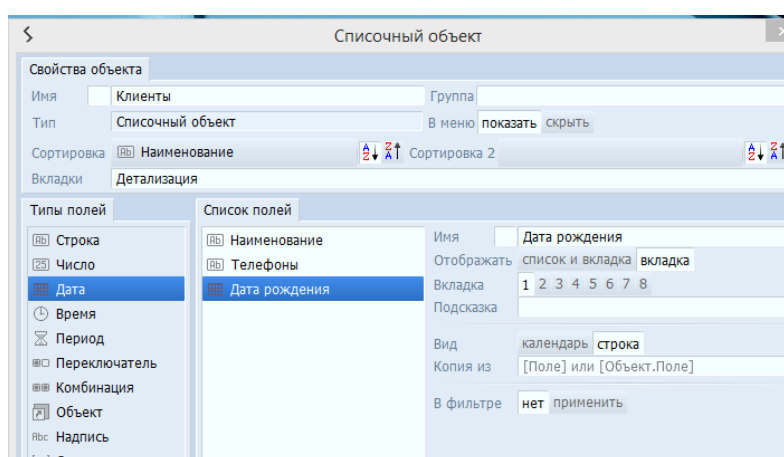


Рис. 6.4. Поле «Дата рождения»

Добавим поле «Переключатель» и введем его имя - «Лицо». В строке «Значения» пишем «Физическое, Юридическое» и включаем фильтр (рис. 6.5).

Добавим поле «Комбинация» и в свойстве поля введем следующее: Имя - «Тип», Значение - «производство, торговля, услуги, другое».



Далее включим фильтр, который будет работать по условию «логическое И». Сохраним все это (рис. 6.6).

Свойства объекта

Имя: Клиенты      Группа:      В меню: показать скрыть

Тип: Списочный объект      Сортировка: Наименование      Сортировка 2:      Вкладки: Детализация

Типы полей

Список полей

Имя: Лицо

Отображать: список и вкладка      вкладка

Вкладка: 1 2 3 4 5 6 7 8

Подсказка:      Значения: Физическое, Юридическое

Вид: строка      список

Стиль: текст      рисунки

В фильтре: нет      применить

Исходное:      Параметры: Передаваемый: из формы

Принимаемый: для фильтра

Рис. 6.5. Поле «Лицо»

Свойства объекта

Имя: Клиенты      Группа:      В меню: показать скрыть

Тип: Списочный объект      Сортировка: Наименование      Сортировка 2:      Вкладки: Детализация

Типы полей

Список полей

Имя: Тип

Отображать: список и вкладка      вкладка

Вкладка: 1 2 3 4 5 6 7 8

Подсказка:      Значения: производство, торговля, услуги, другое

В фильтре: нет      применить

Условие: логическое И      логическое ИЛИ

Рис. 6.6. Поле «Тип»

Далее добавим «Вид деятельности», сделаем его в другом объекте. В нем будет только одно наименование, по которому можно будет фильтровать записи. Включим сортировку.

Объект «Виды деятельности» будет редактироваться не часто, поэтому поместим его в группу «Справочники» и сохраняем (рис 6.7).

Свойства объекта

Имя: Виды деятельности      Группа: Справочники      В меню: показать скрыть

Тип: Списочный объект      Сортировка: Наименование      Сортировка 2:      Вкладки: Детализация

Типы полей

Список полей

Имя: Наименование

Отображать: список и вкладка      вкладка

Вкладка: 1 2 3 4 5 6 7 8

Подсказка:      Подтип: Строка

Исходное:      Маска:      Уникальное: нет      применить

Автоподбор: нет      применить

Ширина колонки: 0 (0 - авто)

Копия из: [Поле] или [Объект.Поле]

В фильтре: нет      применить

Рис. 6.7. Объект «Виды деятельности»

Добавим новый объект «Города», который тоже поместим в группу «Справочники».

Создадим строку «*Название*», включим сортировку и сохраним (рис. 6.8).

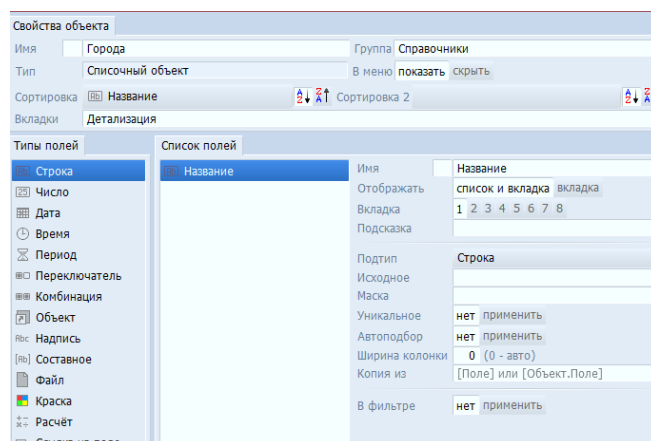


Рис. 6.8. Объект «Города»

Возвращаемся к объекту «*Клиенты*».

Добавляем поле ссылки «*Объект*». В строке «*Источник*» выбираем «*Виды деятельности*», в строке «*Имя*» прописываем «*Деятельность*», и включаем фильтр.

Добавим дополнительную информацию по клиентам, для этого добавим поле «*Составное*» и в строке «*Имя*» прописываем «*Дополнительная информация*» (рис 6.9)

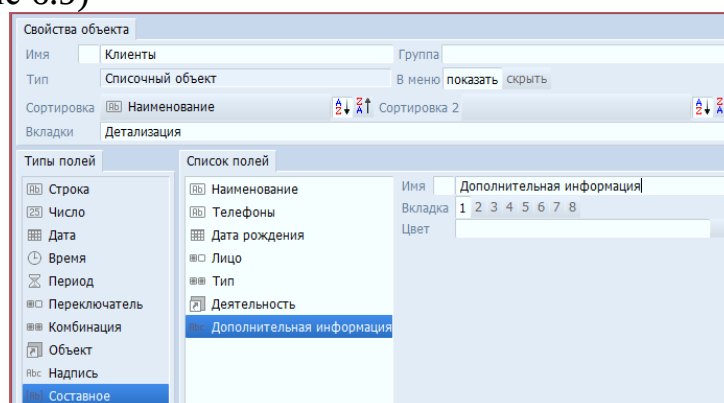


Рис. 6.9. Поле «Дополнительная информация»

Из списка полей перенесем телефоны в Дополнительную информацию. Далее добавим «*Объект*». В строке «*Источник*» выберем «*Города*», а в строке «*Имя*» пропишем – «*Город*» (рис. 6.10).

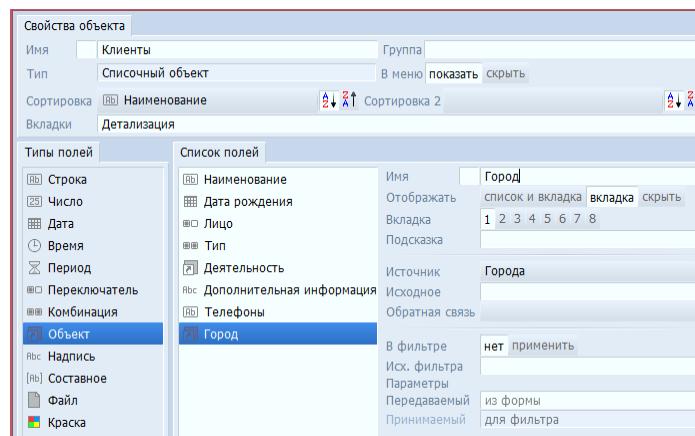


Рис. 6.10. Объект «Город»

Добавим поле «*Строка*», в строке «*Имя*» прописываем «*Адрес*» а в строке «*Подтип*» выбираем «*Карта*»

Добавим поле «*Строка*», в подтипе выбираем «*Email*» и в имени прописываем «*Email*».

Добавим поле «*Строка*», в подтипе выбираем «*Сайт*» и в имени прописываем ссылку на веб-страницу «*WWW*».

И в завершении добавим текстовое поле примечаний: выбираем поле «*Строка*», в имени прописываем «*Примечания*», а в подтипе выбираем «*Текст*». И все сохраняем (рис. 6.11).

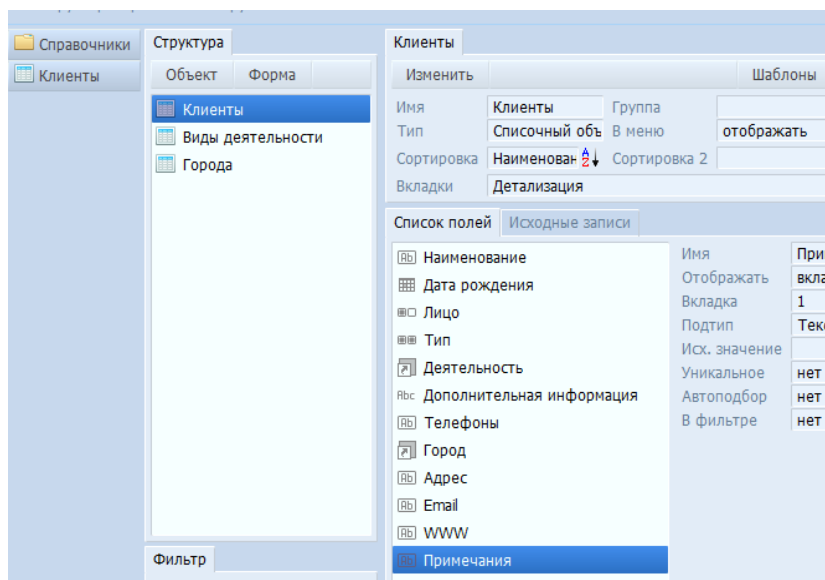


Рис 6.11. Объект «Клиенты»

Откроем «*Справочники*» - объект «*Вид деятельности*» и добавим запись «*Строительство*» (рис 6.12).

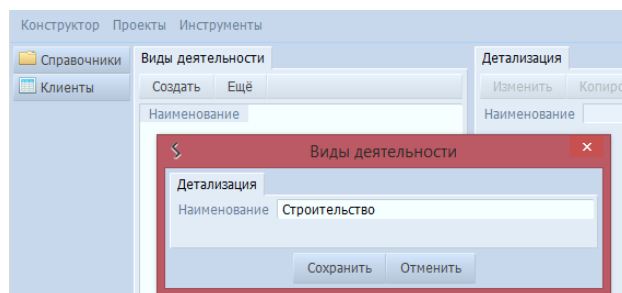


Рис 6.12. Добавление записи в объекте «Виды деятельности»

Аналогично добавим записи: *Отделка*, *Ландшафт*, *Дизайн*, *Сантехника*, *Газовое оборудование*.

Далее для объекта «*Города*» аналогично добавляем записи: *Киев*, *Минск*, *Москва*.

Теперь можно добавить клиентов: «*Клиенты*» - «*Создать*» и заполняем открывшееся окно (рис 6.13).

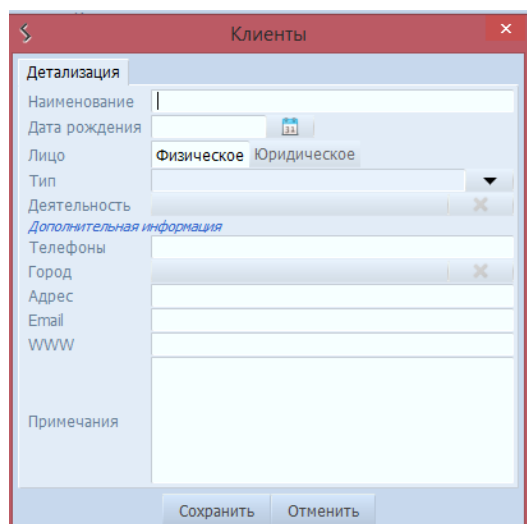


Рис. 5.13. Окно «Клиенты»

При работе с данными объекта можно перейти к другим объектам, на которые имеется ссылка (рис. 6.14).

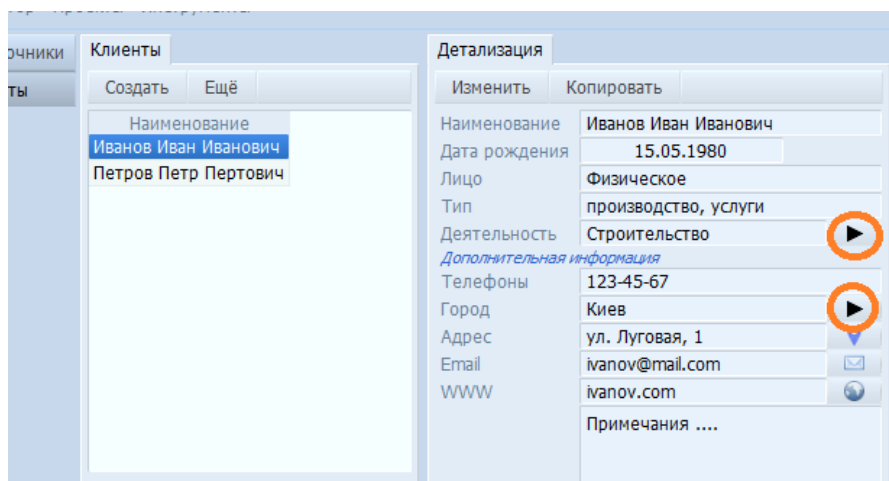


Рис 5.14. Ссылка на объекты

Можно открыть адрес в картографическом сервисе. Если их не хватает, то можно настроить карты: «Инструменты» - «Настройки» - «Карты» - добавляем необходимое – «Сохранить» (рис. 6.15).

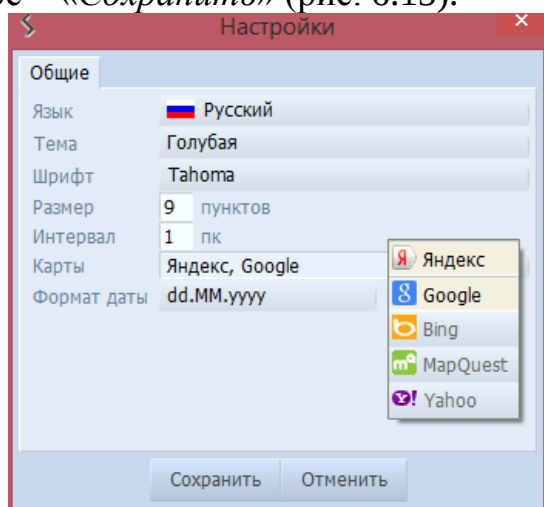


Рис. 6.15. Создание картографии

Так же можно отправить письмо по указанному адресу или открыть ссылку на сайт.

Создадим объект «Контакты»: «Конструктор» - «Структура» - «Объект».

Сначала добавим справочник «Вид контакта». В нем определим поле «Название» (тип поля «Строка») и поле «Порядок» (тип поля «Число»).

Для поля «Порядок» в строке «Исходное» выбираем «макс.+1», сортировку задаем по Порядку, помещаем объект в Справочники и сохранить (рис. 6.16)

The screenshot shows the 'Свойства объекта' (Object Properties) dialog box. The 'Имя' (Name) is 'Вид контакта' and the 'Группа' (Group) is 'Справочники'. The 'Тип' (Type) is 'Списочный объект' (List object). The 'Сортировка' (Sorting) is set to 'Порядок' (Order). The 'Вкладки' (Tabs) section shows 'Детализация' (Details). The 'Типы полей' (Field types) list on the left includes 'Число' (Number), which is selected. The 'Список полей' (List of fields) shows 'Порядок' (Order) as the first field. The 'Имя' (Name) of the field is 'Порядок'. The 'Отображать' (Display) option is 'список и вкладка' (list and tab). The 'Вкладка' (Tab) is set to '1'. The 'Подсказка' (Tooltip) is empty. The 'Знаков' (Characters) field is set to '0' with the note 'после запятой' (after comma). The 'Суффикс' (Suffix) is empty. The 'Исходное' (Original) value is 'значение макс.+1 последнее' (value max.+1 last). The 'Зависит' (Depends) field is empty. The 'Значение' (Value) is '0'. The 'Копия из' (Copy from) field is '[Поле] или [Объект.Поле]' (Field or Object.Field).

Рис. 6.16. Объект «Вид контакта»

Теперь создадим сам объект «Контакты». Первым полем будет «Дата», в котором применяем фильтр.

Для строгой очередности зададим нумерация: в поле «Число» введем имя «№», включим автогенерацию (исходное – «макс.+1») и зададим сортировку по ней по убыванию (рис. 6.17).

The screenshot shows the 'Свойства объекта' (Object Properties) dialog box. The 'Имя' (Name) is 'Контакты' and the 'Группа' (Group) is empty. The 'Тип' (Type) is 'Списочный объект' (List object). The 'Сортировка' (Sorting) is set to 'Порядок' (Order). The 'Вкладки' (Tabs) section shows 'Детализация' (Details). The 'Типы полей' (Field types) list on the left includes 'Число' (Number), which is selected. The 'Список полей' (List of fields) shows 'Дата' (Date) as the first field. The 'Имя' (Name) of the field is '№'. The 'Отображать' (Display) option is 'список и вкладка' (list and tab). The 'Вкладка' (Tab) is set to '1'. The 'Подсказка' (Tooltip) is empty. The 'Знаков' (Characters) field is set to '0' with the note 'после запятой' (after comma). The 'Суффикс' (Suffix) is empty. The 'Исходное' (Original) value is 'значение макс.+1 последнее' (value max.+1 last). The 'Зависит' (Depends) field is empty. The 'Значение' (Value) is '0'. The 'Копия из' (Copy from) field is '[Поле] или [Объект.Поле]' (Field or Object.Field).

Рис. 6.17. Создание нумерации

Добавим клиента: Тип поля «Объект» - Источник «Клиенты» - отображать «список и вкладка»- В фильтре «Применить» (рис 6.18).

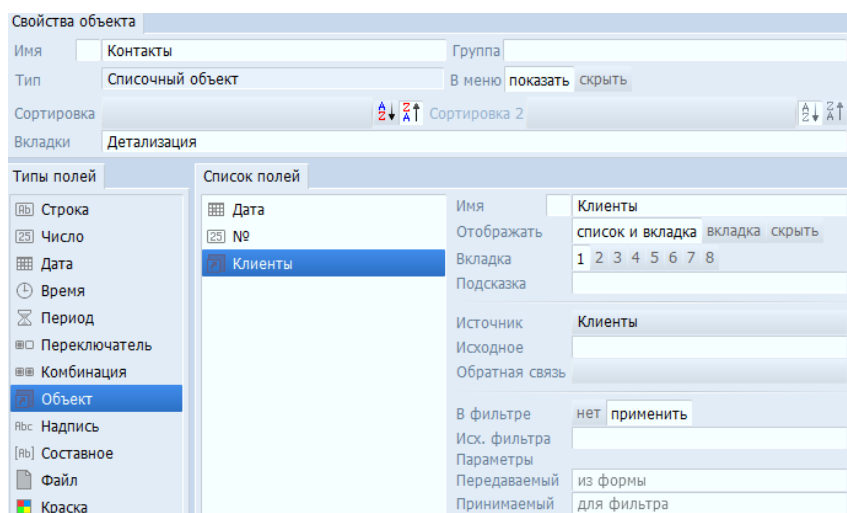


Рис. 6.18. Создание поля «Клиенты»

Создадим поле «Вид контакта» (тип поля «Объект»). Где *источник* – «Вид контакта», *отображать* – «список и вкладка».

Далее добавим поле «Строка», запишем в строке *Имя* - «Результат», выберем *подтип* – «Текст». И все сохраняем.

Открываем «Справочники» - «Вид контакта» и добавим записи «Входящий звонок», «Исходящий звонок», «Встреча», «Входящее письмо», «Исходящее письмо» (рис 6.19).

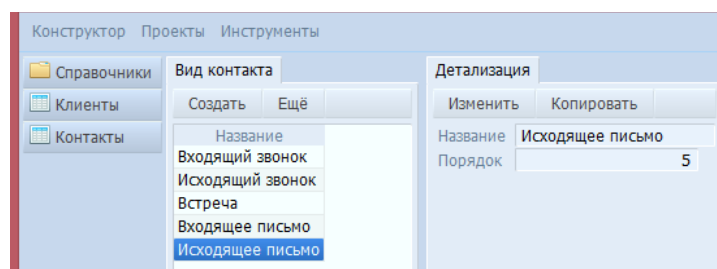


Рис. 5.19. Объект «Вид контакта»

Далее открываем «Контакты», создаем записи и заполняем появившееся окно.

**Задание 6.1.** Используя теоретический и практический материал лабораторной работы №5, создать БД, состоящую из нескольких таблиц (не менее трёх).

Варианты заданий:

1. Абитуриент
2. Ателье
3. Библиотека
4. Биржа труда
5. Больница
6. Гараж
7. Гастроном
8. Деканат вуза
9. Детский лагерь
10. Санаторий
11. Речной порт
12. Прокат видеокассет
13. Поликлиника
14. Музыкальная коллекция
15. Конноспортивная школа
16. Книжный магазин
17. Кафе
18. Избирательная комиссия
19. Детский сад
20. Семейный бюджет

## Лабораторная работа № 7. «Типы связей таблиц с дочерним объектом в форме»

Форма связи позволяет установить подчинённые или равнозначные отношения между различными объектами, один из которых является родительским, а остальные — дочерними.

**Дочерние объекты** добавляются на вкладке *Дочерние объекты*. Добавление производится с указанием типа связи: подчинённая или дочерняя. При этом в структуре автоматически создаются системные объекты/поля (с префиксом ~, серого оттенка и скрытые от отображения) реализующие выбранную связь, а при работе с данными — под вкладками детализации родительского объекта добавляются вкладки с записями дочерних объектов, которые связаны с выбранной записью родительского объекта.

**Подчинённая связь** характеризуется привязкой множества записей дочернего объекта к одной записи родительского (один-ко-многим). При этом запись дочернего объекта может быть привязана только к одной записи родительского.

**Равнозначная связь** характеризуется привязкой множества записей одного объекта ко множеству записей другого (много-ко-многим). При этом любая запись дочернего объекта может быть привязана к любой записи родительского.

Для объекта могут быть определены записи, которые автоматически будут созданы в нём, если он задан как дочерний для другого объекта в форме связи. Автоматическое добавление записей в дочерний объект происходит при создании записи в родительском объекте на форме связи. Добавляемые записи в дочернем объекте имеют привязку к созданной пользователем записи родительского объекта через системное поле ссылки.

### 7.1. Создание равнозначной связи

Создадим новую базу данных «*Форма связи*».

Создадим объект «*Предметы*», добавим поля *Наименование* (тип поля *строка*) и *Дата* (тип поля — *дата*) (рис 6.1). Сохраняем получившийся результат.

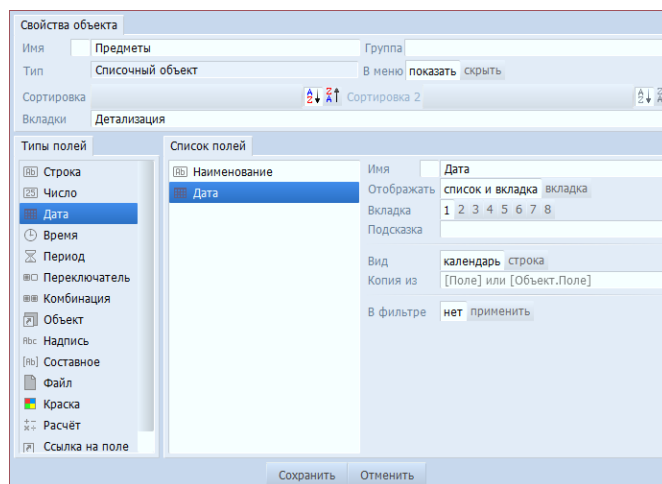


Рис. 7.1. Объект «Предметы»

Создаем следующий объект «Студенты» с полями – «ФИО», «Телефон» (тип поля - *Строка*) (рис. 7.2). Сохраняем результат.

Рис. 7.2. Объект «Студенты»

Возвращаемся к объекту «Предметы» и создаем пару дисциплин: *Математика, Физкультура, История, Философия, Экология* (рис. 7.3).

| Наименование | Дата       |
|--------------|------------|
| Математика   | 17.11.2015 |
| Физкультура  | 20.11.2015 |
| История      | 23.11.2015 |
| Философия    | 26.11.2015 |
| Экология     | 29.11.2015 |

Рис. 7.3. База «Предметы»

Далее создаем участников (рис 7.4).

| ФИО            | Телефон  |
|----------------|----------|
| Иванов Иван    | 11-11-11 |
| Петров Петр    | 22-22-22 |
| Сидоров Иван   | 33-33-33 |
| Краснов Андрей | 44-44-44 |
| Иванова Ирина  | 55-55-55 |
| Данилова Оля   | 66-66-66 |

Рис. 7.4. База «Студенты»

Теперь свяжем эти два объекта между собой, чтобы студентов можно было привязать к изучаемым предметам.

Для этого переходим в режим *Конструктор – Структура – Предметы – Форма связи* (рис 7.5)

Рис. 7.5. Выбор формы связи

Создаем форму на базе предметов (объект – *Предметы*), добавляем поля для изображений в списке и сохраняем (рис 7.6)



7.6. Создание связи

Теперь форма *Предметы* практически ничем не отличается от объекта *Предметы*

Разместим подчиненный объект с равнозначной связью: *Конструктор – Структура – Дочерние объекты*. Добавляем дочерние объекты *Студенты*, устанавливаем связь – *равнозначная*. Сохраняем (рис. 7.7).

Рис. 6.7. Создание дочерних объектов

Появился системный объект с двумя ключевыми полями, которые являются ссылками на нужный нам объект: *Предметы*, *Студенты* (рис.7.8).

Рис. 7.8.

Открываем форму *Предметы* и добавляем студентов к каждому предмету (проделываем для каждого предмета) (рис. 7.9)

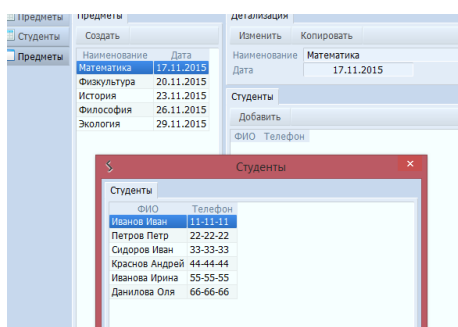


Рис. 7.9.

Таким образом сделали форму для *Предметов*.

Создайте самостоятельно подобную форму с участием объекта *~Предметы-Студенты* для *Студентов*.

Добавим дополнительное поле, характеризующее связь *~Предметы-Студенты*.

Создадим объект *Роль*, содержащий поле *Строка* с именем *Название*. Сохранить.

В *Роль* добавим варианты: *Студенты*, *Преподаватели* (рис. 7.10)

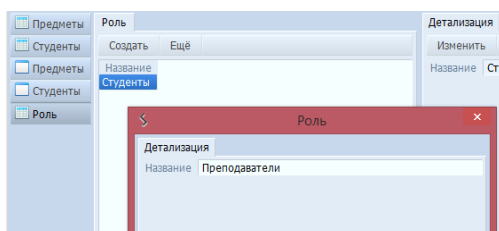


Рис. 7.10.

Теперь для связующего объекта *~Предметы-Студенты* добавим поле *объект*, источником которого будет объект *Роль* (рис. 7.11). Делаем его видимым: для этого выбираем свойство поля *отображать* - список и вкладка. Сохранить.

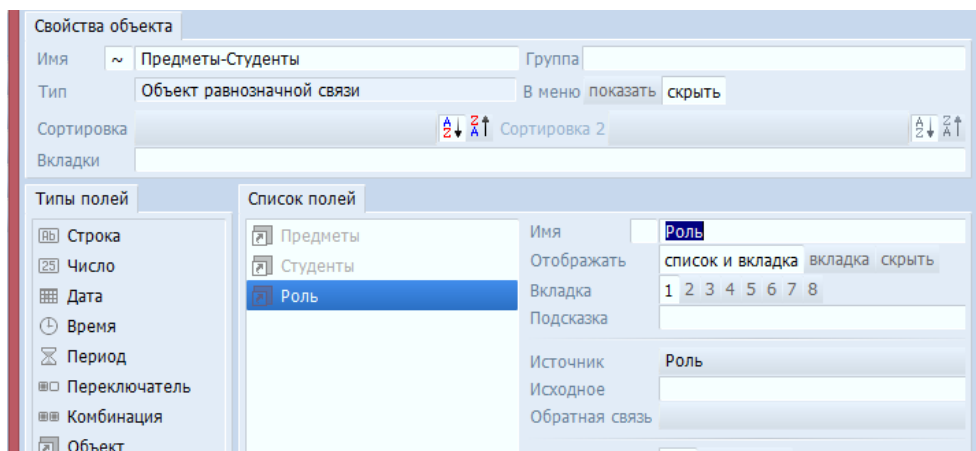


Рис. 7.11.

Теперь у нас появилось дополнительное поле в обоих объектах. При добавлении (или изменении) записи связующего объекта открывается дополнительное окно, в котором указаны дополнительные поля (рис.7.12).

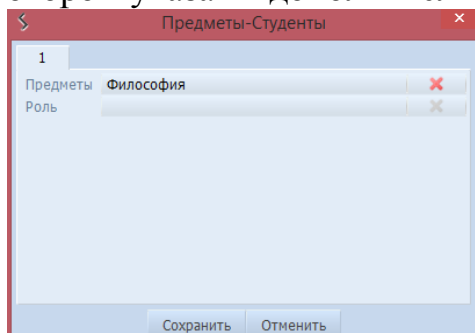


Рис.7.12

## 7.2. Создание подчиненной связи

Для примера создадим новую базу «Подчиненная связь».

Добавим объект *Заказы*, в который добавим поле *Число* (*имя* - №, *исходное* – макс.+1) и поле *Дата* (*отображать* – список вкладка). Сохранить.

Заполним объект *Заказы* (таблица 6.1):

Таблица 7.1

| № | Дата       |
|---|------------|
| 1 | 29.11.2015 |
| 2 | 30.11.2015 |
| 3 | 1.12.2015  |

Создаем следующий объект *Услуги*. Добавим следующие поля: *Строка* (*имя* – Наименование), *Число* (*имя* – Стоимость, *отображать* – список и вкладка). Сохранить.

Теперь сделаем объект *Услуги* подчиненным объекту *Заказы*.

Создадим форму на базе объекта *Заказы*: *Структура* – *Форма* – *связи*. В свойстве *Объект* выбираем *Заказы* и добавляем все поля для отображения в списке. Сохранить.

И теперь здесь нужно разместить подчиненный объект. Выбираем вкладку *Дочерние объекты* – *Добавить*. В появившемся окне в качестве объекта выбираем *Услуги*, связь – *подчиненная*.

Получили форму родительского объекта *Заказы* и на ней есть дочерний объект *Услуги* (рис. 7.13).

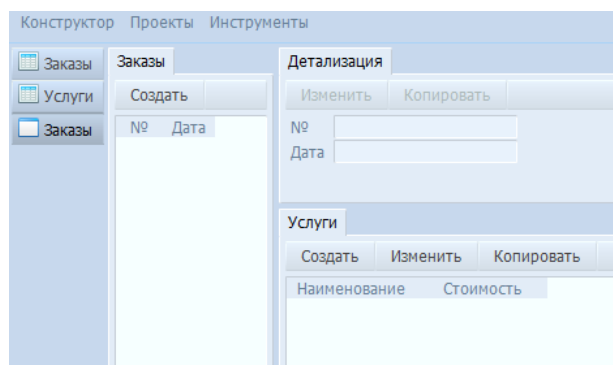


Рис. 7.13.

Заполняем объект родительский объект *Заказы*:

- для заказа №1: помыть полы – 100, выбить ковер – 300;
- для заказа №2: вынести ведро – 500, помыть посуду – 75, сложить книги – 15;
- для заказа №3: протереть пыль – 200, постирать вещи – 700.

У объекта *Услуги* появилась скрытая форма *Заказы*, которая является ссылкой на объект *Заказы*. Образуется связь *один ко многим*.

Теперь рассмотрим вариант, когда нужно выбирать варианты.

Создадим объект *Товары*. Добавим поле *Строка* (имя – *Название*). Из этого объекта можно будет выбирать варианты для заказов. Заполним объект *Товары*: щетка, тряпка, мыло, порошок, руки.

Создадим промежуточный объект *Состав товара*. Первое его поле будет *Объект* (источник – *Товары*), второе – *Число* (имя – *Количество*, отображать – *список и вкладка*). Сохранить.

Теперь в форме *\_Заказы* добавляем дочерний объект *Состав товара*.

Открываем форму *Заказы* и видим, что добавилась вторая вкладка *Состав товара* (рис. 6.14)

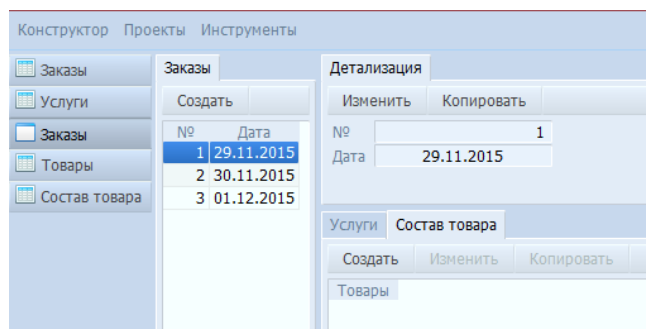


Рис. 7.14.

Заполняем самостоятельно *Состав товара* для каждого заказа.

**Задание 7.1.** Используя теоретический и практический материал лабораторной работы №7, создайте подчиненную и равнозначную связь по своему варианту (см. лабораторную работу №6).

## Лабораторная работа № 8. «Формирование шаблонов вывод списка записей в документ Microsoft Word »

Создадим небольшую базу данных, состоящую из двух объектов: *Клиенты* и *Заказы*.

Объект *Клиенты* содержит два поля: ФИО, Контакт (рис. 8.1).

Рис. 8.1

Объект *Заказ*: № (тип поля – число), дата (тип поля – дата), клиент (тип поля – объект), состояние (тип поля – переключатель, значения - заказ, обрабатывается, выполнен ) (рис. 8.2)

Рис. 8.2.

Для получения отчета с нужными полями из списка, их последовательно и стилями служат шаблоны в формах.

На базе объекта *Заказ* создаем форму и указываем какие поля будут добавлены в список (№, дата, клиент), также можно изменить их последовательность (рис. 8.3). Сохранить.

Рис. 8.3

Рассмотрим самый простой способ создания списочного отчета по записям формы. Для этого в структуре на форме *\_Заказы* отключаем вкладку с данными которые на ней отображаются (рис 8.4) и получаем форму без детализации.

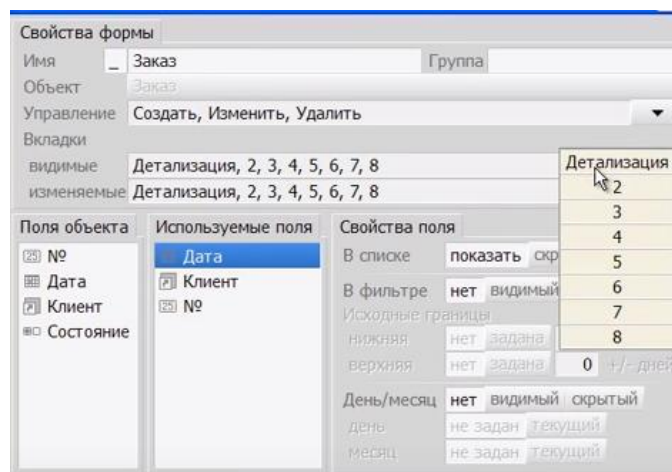


Рис. 8.4

На базе такой формы создаем шаблон под именем *Список* (рис. 8.5), в котором создается структура для экспорта списка только записей без детализации (рис. 8.6).

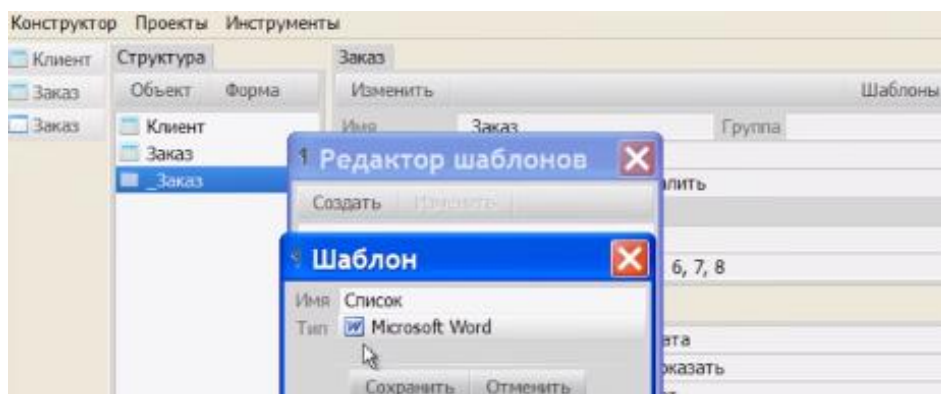


Рис. 8.5

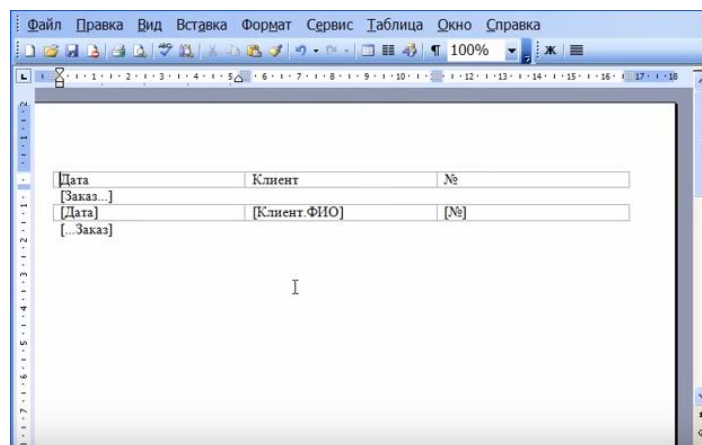


Рис. 8.6

Закрываем появившееся окно.

Далее открываем форму *Заказ - Документы* и выбираем из списка документ *Список*. (рис 8.7) и смотри, что получилось.

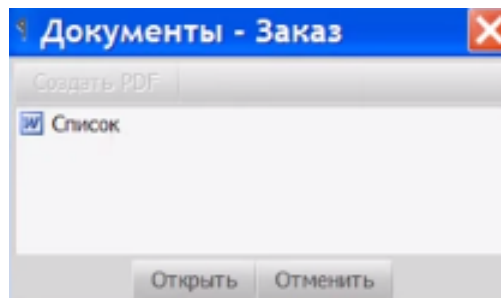


Рис. 8.7

Шаблон документа можно оформить так как вам нужно (поменять ширину столбцов, изменить шрифт, размер и т.д.).

Точно таким же образом можно создать шаблон на базе Excel. Создайте самостоятельно шаблон по своим требованиям.

Рассмотрим как создавать шаблон с подчиненными списками записи. Для этого возвращаем видимую вкладку для формы *\_Заказ* (рис 8.8).

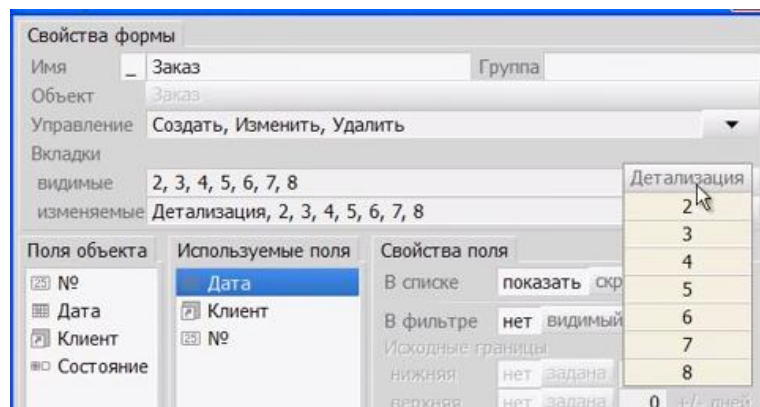


Рис. 8.8

Добавим новый объект *Товары*, который включает в себя поле *Строка* (имя – *Наименование*) и объект *Услуги*, который также включает в себя поле *Строка*, с именем *Наименование*.

Далее на форме *\_Заказ* добавляем эти объекты в качестве дочерних и подчиненных.

Теперь у нас есть форма, в которой есть весь список заказов и для каждой записи из заказа есть список товаров и услуг, которые ему соответствуют.

Заполним в форме *\_Заказы* вкладку *Товар*, как показано на рис. 8.9 и вкладку *Услуги* (рис. 7.10) для каждого клиента из списка.

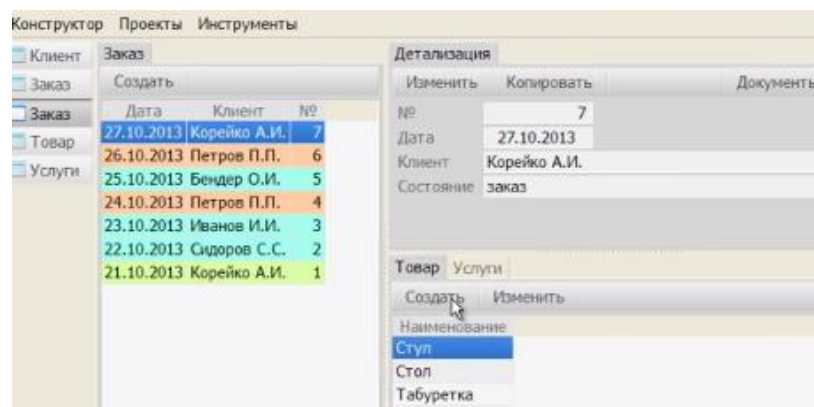


Рис. 8.9.



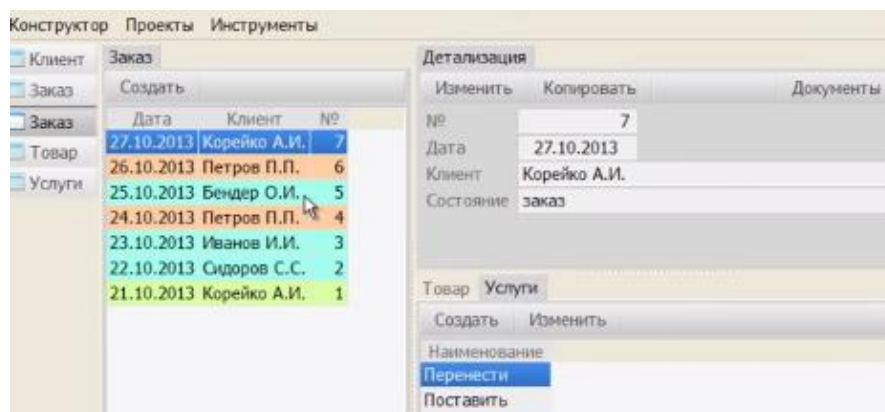


Рис. 8.10

Далее создадим шаблон под именем *Документ со списками*, в котором будут списки товаров и услуг (рис. 7.11)

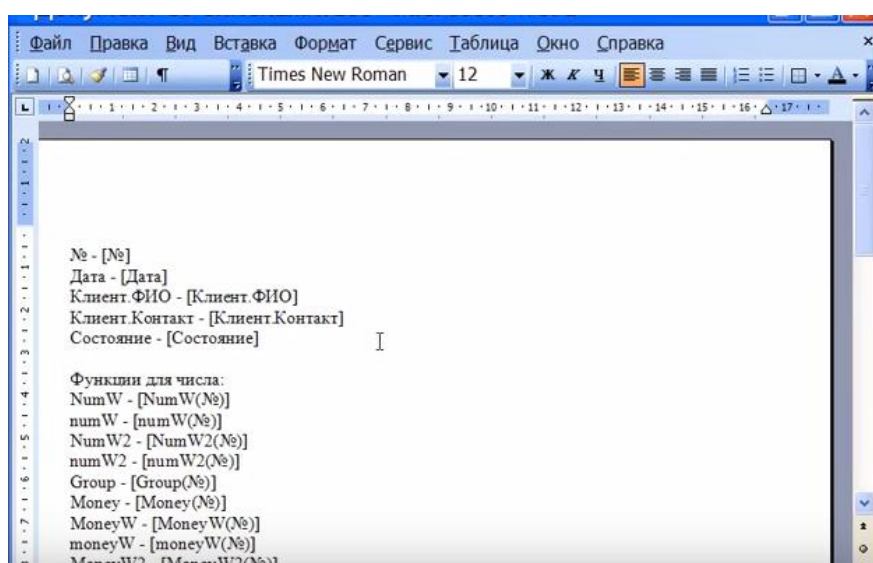


Рис. 8.11

Удаляем лишнее и оставляем данные по заказу и структуру документа, предназначенную для вывода списков как товаров, так и услуг. Закрываем документ.

Проверяем результат. Заходим в форму *\_Заказ* – вкладку *Документы*, выбираем шаблон *Документ со списками* и смотрим на получившийся результат (рис. 8.12)



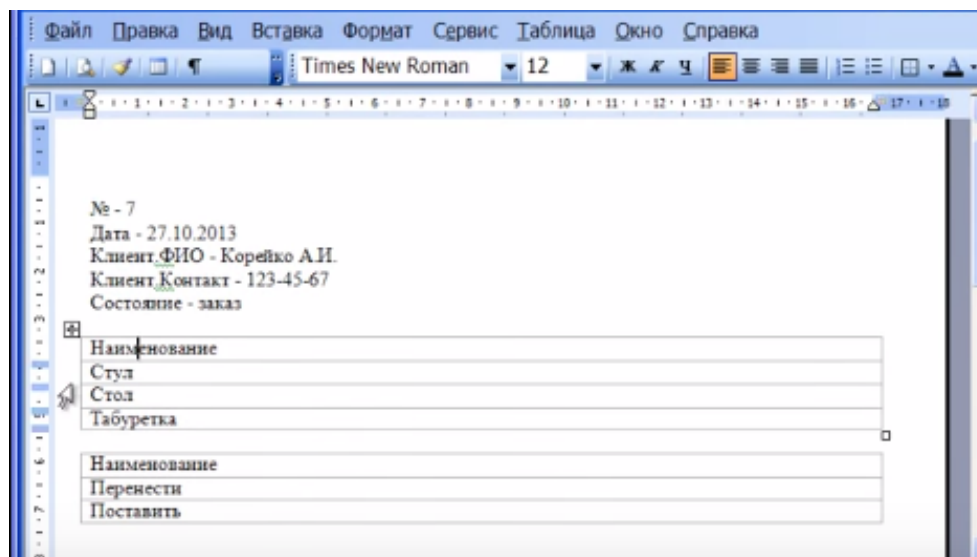


Рис. 7.12

**Задание 8.1.** Используя теоретический и практический материал лабораторной работы №8, создайте шаблоны документов по своему варианту (см. лабораторную работу №6).