

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ

НАБЕРЕЖНОЧЕЛНИНСКИЙ ИНСТИТУТ (ФИЛИАЛ) ФЕДЕРАЛЬНОГО
ГОСУДАРСТВЕННОГО АВТОНОМНОГО ОБРАЗОВАТЕЛЬНОГО
УЧРЕЖДЕНИЯ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

КАФЕДРА «БИЗНЕС-ИНФОРМАТИКА И МАТЕМАТИЧЕСКИЕ МЕТОДЫ
В ЭКОНОМИКЕ»



Работа с СУБД Microsoft Access и SQL Server в среде Delphi

Электронный образовательный ресурс

**Набережные Челны
2018**

Работа с СУБД Microsoft Access и SQL Server в среде Delphi : Электронный образовательный ресурс для студентов направлений подготовки «Бизнес-информатика» и «Прикладная информатика (в экономике)» / Составитель – А.Г.Исавнин. Набережные Челны: 2018, - 39 с.

**Рецензенты: д.т.н., профессор А.К.Розенцвайг,
к.т.н., доцент Р.И.Мулюков.**

Электронный образовательный ресурс содержит подробное пошаговое описание процесса создания таблиц форматов MS Access и MS SQL Server, организации доступа к данным через клиентское приложение средствами Delphi с использованием механизма ActiveX Data Objects. Для студентов направлений подготовки «Бизнес-информатика» и «Прикладная информатика (в экономике)»

СОДЕРЖАНИЕ

Содержание	3
Введение	4
Создание таблиц в СУБД Microsoft SQL Server	5
Подключение таблиц SQL Server к компонентам Delphi ...	15
Разработка запросов	21
Создание таблиц в СУБД Microsoft Access 2000	25
Подключение таблиц MS Access к компонентам Delphi ...	30
Связь таблиц	33
Организация связи в Delphi	35
Рекомендуемая литература	39

ВВЕДЕНИЕ

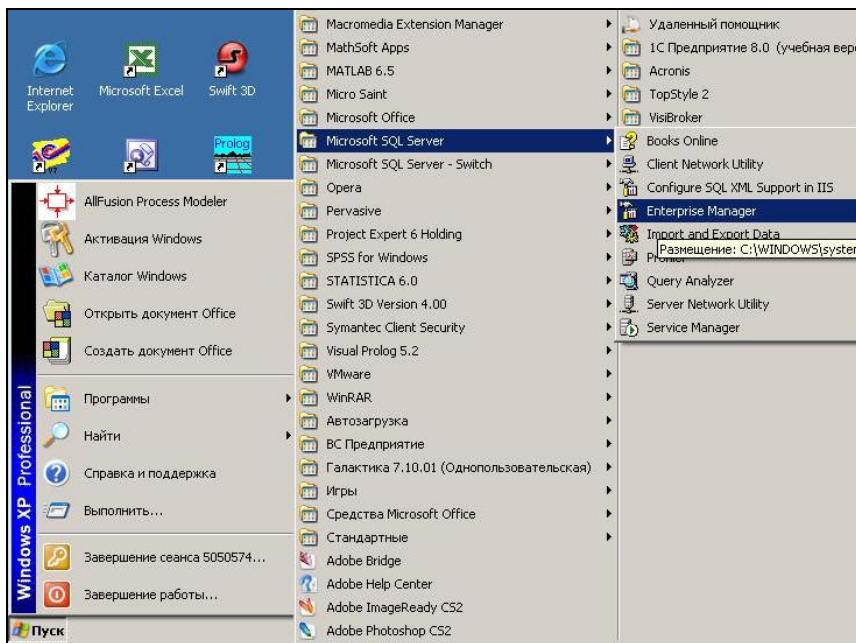
В настоящем пособии представлены способы создания клиентских приложений, обеспечивающих работу с базами данных форматов Microsoft Access и Microsoft SQL Server (2000). В качестве механизма организации доступа к данным предлагается использовать технологию компании Microsoft ActiveX Data Objects (ADO). Эта технология представляет собой высокоуровневую объектную надстройку над OLE DB. Использование ADO является альтернативой Borland Database Engine (BDE), обеспечивающей более эффективную работу с данными. Возможность обращаться к данным посредством ADO появилась еще в пятой версии Delphi.

Для работы с ADO предусмотрены компоненты, расположенные на страничке ADO палитры компонентов Delphi (ADOConnection, ADOTable, ADOQuery и др.). Связь с базой данных в технологии ADO осуществляется обычной цепочкой : набор данных, источник данных (компонент DataSource), компоненты управления и отображения данных (DBGrid, DBEdit и др.). Отличие заключается только в первом звене этой цепочки, в котором вместо компонентов, расположенных на страничке BDE палитры компонентов Delphi, используются компоненты, расположенные на вкладке ADO.

Большинство компонентов, предназначенных для работы с ADO, аналогичны компонентам, работающим с BDE. Например, компонент ADOTable соответствует компоненту Table, компонент ADOQuery – компоненту Query, компонент ADOConnection – компоненту Database и т.д.

СОЗДАНИЕ ТАБЛИЦ В СУБД MICROSOFT SQL SERVER

1. Запускаем СУБД, выполняя команды: Пуск - Программы - Microsoft SQL Server - Enterprise Manager



2. Открывается окно Enterprise Manager:



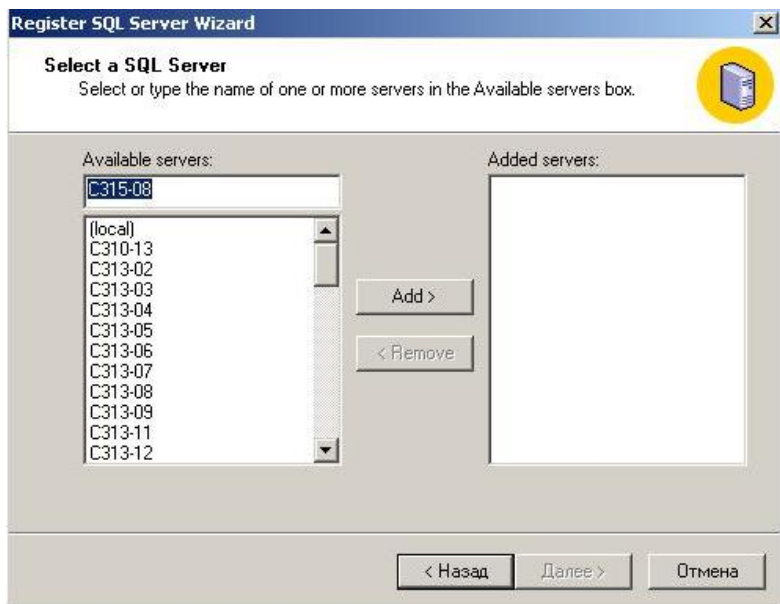
3. Для того чтобы иметь доступ к удаленным базам данных, необходимо зарегистрироваться. Регистрация осуществляется следующим образом: Правой кнопкой мыши нажимаем на значок SQL Server Group и выполняем команду контекстного (всплывающего) меню New SQL Server Registration



После выполнения данной команды открывается первое диалоговое окно мастера регистрации:



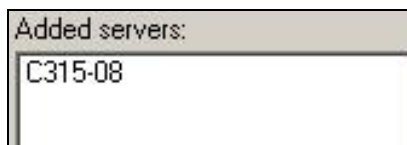
При нажатии на кнопку «Далее» появляется диалоговое окно выбора сервера :



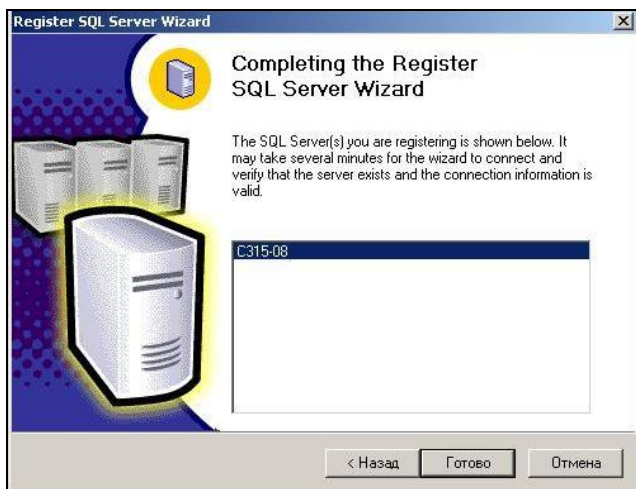
По умолчанию в строке Available servers указывается номер сервера, на котором вы регистрируетесь в данный момент.

Также у вас есть возможность организовать доступ к другим серверам, например C313-02. Данная надпись означает, что вы подключаетесь к компьютеру, находящемуся в 313 аудитории под номером 2. Важно запомнить номер выбранного вами сервера.

После выбора необходимого сервера нажимаем на кнопку ADD>. В результате в поле Added server появляется номер вашего сервера:



После выполнения данной процедуры нажимаем кнопку «Далее >>». Затем, следуя инструкциям мастера и оставляя все настройки по умолчанию, нажимаем кнопку «Далее», до возникновения окна завершения регистрации:



Нажимаем кнопку «Готово», появляется окно сообщений:

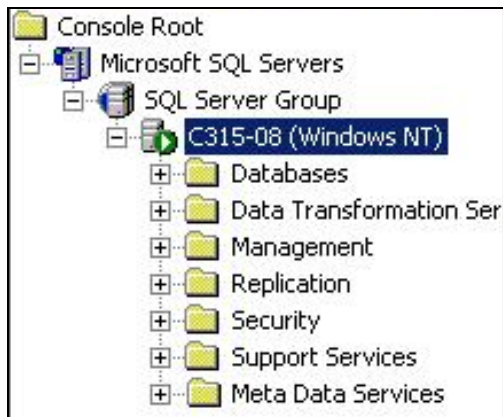


Статус Registered successfully означает, что регистрация прошла успешно. Закрываем это окно. На этом регистрация закончена.

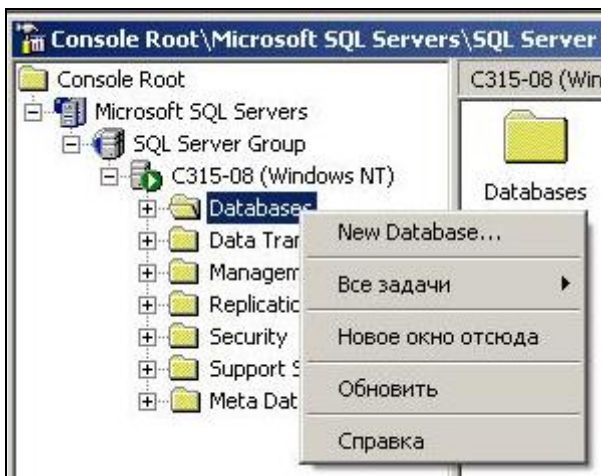
4. Возвращаемся в окно Enterprise Manager.и теперь здесь вы увидите тот сервер, на котором вы зарегистрировались:



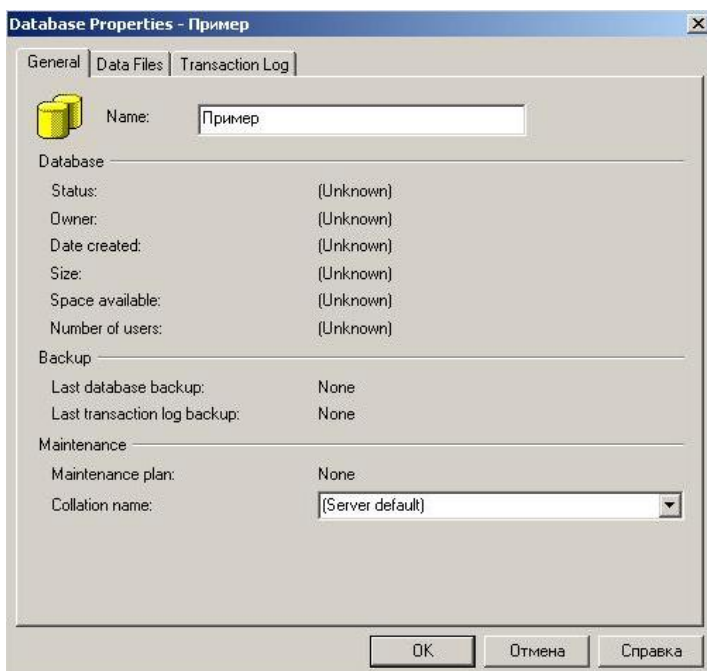
5. Нажмите мышью на знак «+» слева от SQL Server Group. Появляется ваш зарегистрированный сервер. Нажмите мышью на знак «+» возле вашего сервера -появится список доступных папок:



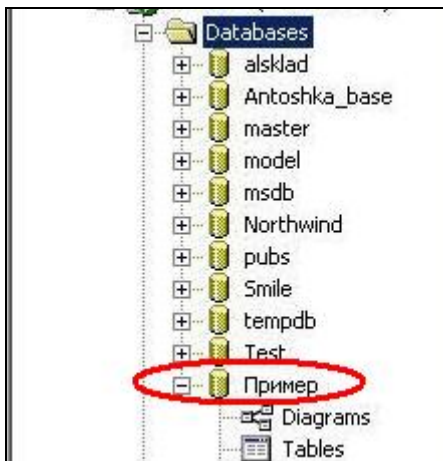
6. Теперь необходимо создать новую базу данных (New Database). Нажимая правой кнопкой мыши на папке Databases, выбираем команду New Database...



7. Появляется диалоговое окно, где в поле Name необходимо ввести название создаваемой базы данных:

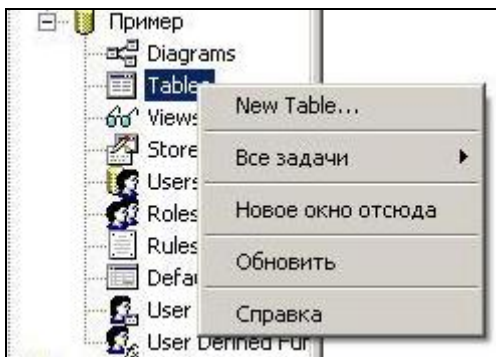


8. Нажимаем на кнопку «Ok», создается новая база данных, которую можно увидеть раскрыв список Databases:

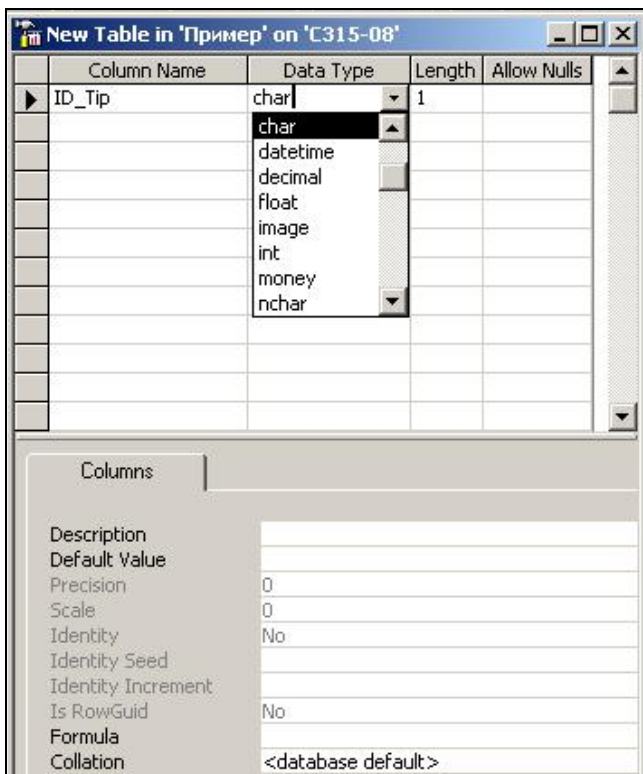


Важно запомнить название созданной вами базы данных !!!

9. После регистрации сервера и создания своей базы данных приступаем к созданию таблиц. Для этого на элементе Tables щелкаем правой кнопкой мыши и выбираем команду New Table (Новая таблица):



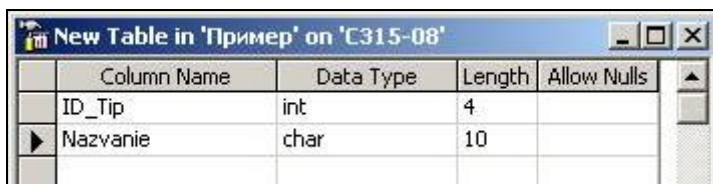
10. Появляется диалоговое окно создания новой таблицы:



В столбце Column Name нужно ввести название поля; в столбце Data Type – указать тип поля; в Length указывается длина поля (для строковых полей); в столбце Allow nulls – ставиться мышью галочка, если в дальнейшем при заполнении таблицы это поле можно оставить пустым.

Внизу можно указать дополнительные свойства для выбранного поля.

После определения всех необходимых полей и их свойств, структура таблицы должна выглядеть следующим образом:



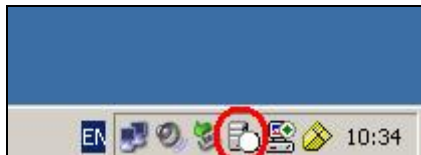
11. При закрытии окна создания таблицы появится следующее диалоговое окно, где необходимо ввести название таблицы (в нашем случае название таблицы - test):



12. При нажатии на кнопку «OK» созданная Вами таблица появится в списке таблиц.

Tables 1 Item				
Name	Owner	Type	Create Date	
test	dbo	User	02.06.2007 10:33:07	

13. Закрываем окно Enterprise manager. На панели задач находим значок сервера.

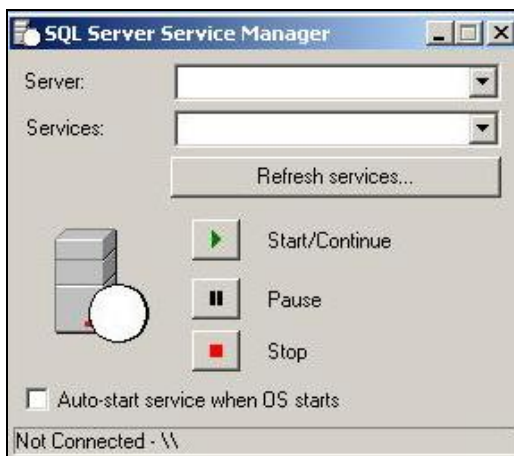


Если в белом кружке этого значка не горит зеленая стрелка (работа с сервером невозможна), то следует проделать следующие операции:

- Нажимаем правой кнопкой мыши на данном значке, появляется контекстное меню, выбираем команду Open SQL Server Service Manager:



- При выборе данной команды появляется следующее диалоговое окно:



Из первого (верхнего) раскрывающегося списка (Server) необходимо выбрать номер своего зарегистрированного сервера. Если вашего сервера в списке не оказалось, то необходимо его номер вписать самостоятельно. Второе поле (Services) следует оставить пустым.



После ввода номера сервера закрываем окно. Теперь вы видите, что в белом кружочке появилась зеленая стрелка. Это означает, что ваш сервер запущен.



Подключение таблиц SQL Server к компонентам Delphi

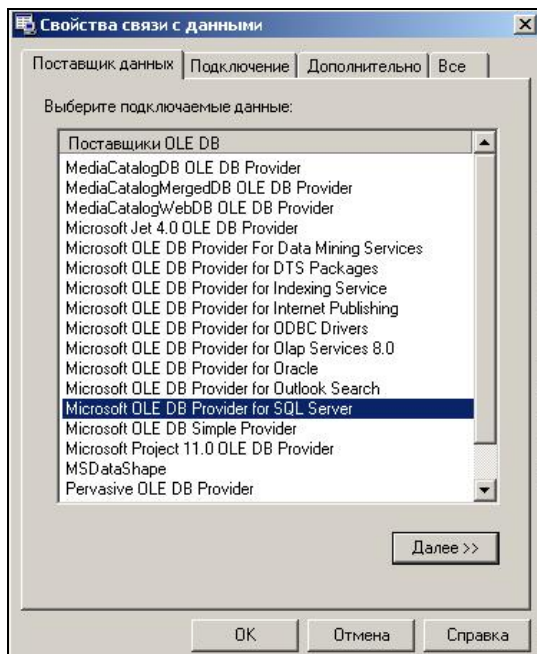
1. Запускаем среду Delphi: Пуск – Программы – Borland Delphi 7 – Delphi 7.
2. Для подключения таблиц SQL Server будем использовать механизм доступа ADO. С вкладки ADO палитры компонентов выбираем компонент ADOTable (Таблица)



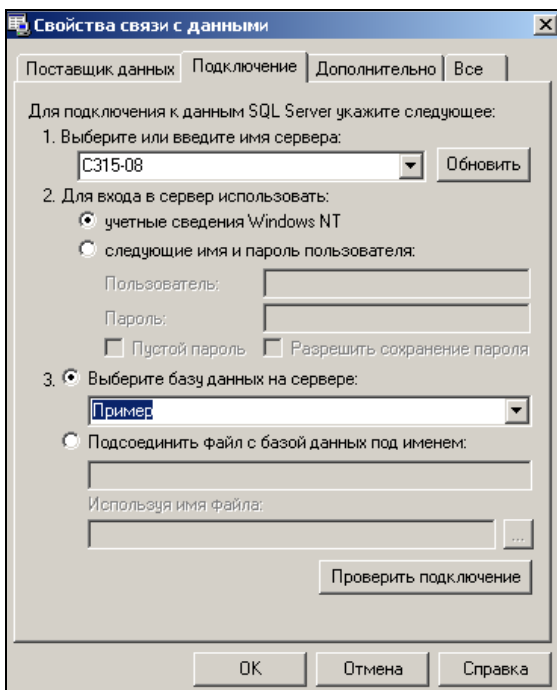
3. В окне инспектора объектов (Object Inspector) в свойствах компонента ADOTable выбираем свойство ConnectionString и нажимаем мышью на кнопку с 3 точками.



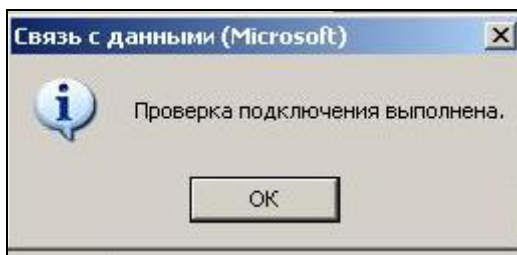
4. Появится диалоговое окно, в котором необходимо нажать на кнопку Build.
5. Откроется диалоговое окно «Свойства связи с данными». Для связи с SQL Server необходимо выбрать из предложенного списка «Microsoft OLE DB Provider for SQL Server» и нажать на кнопку «Далее».



6. В следующем диалоговом окне необходимо выбрать название сервера, установить флажок на учетные сведения Windows NT и выбрать свою созданную базу данных.



7. После того, как все параметры выбраны, необходимо нажать на кнопку «Проверить подключение» - должно появиться сообщение:



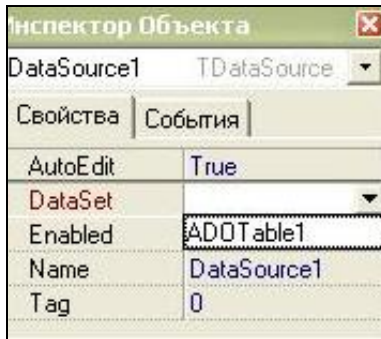
8. Закрываем все окна, нажимая кнопку “OK”. В результате в свойстве компонента ADOTable в строке ConnectionString появляется путь к вашей базе данных.
9. В свойстве TableName компонента ADOTable следует указать имя таблицы, выбрав его из раскрывающегося списка



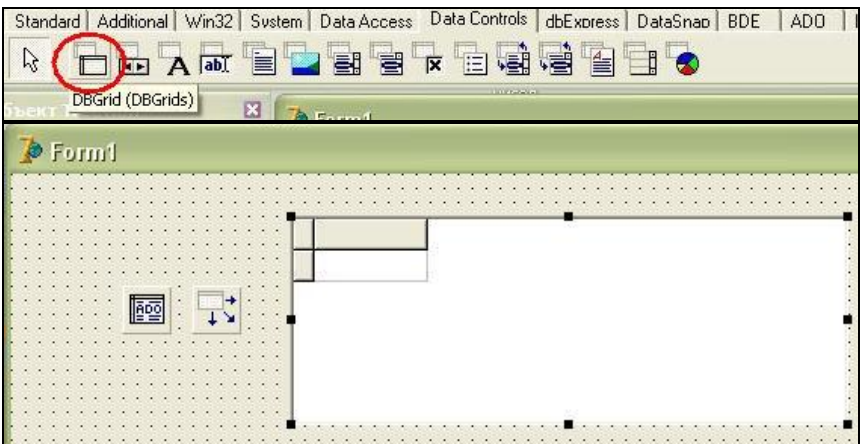
10. В свойстве Active компонента ADOTable выбираем значение True.
11. Располагаем на форме невидимый компонент DataSource (Источник данных) с вкладки Data Access палитры компонентов.



12. В свойстве DataSet компонента DataSource выбираем ADOTable1.



13. Размещаем компонент DBGrid с вкладки Data Controls на форме. В отличие от компонентов ADOTable и DataSource, которые являются невидимыми, компонент DBGrid является визуальным, то есть при запуске приложения он будет отображаться на форме.



14. В свойстве DataSource компонента DBGrid выбираем соответствующий источник данных (DataSource1):



15. Для заполнения таблицы, необходимо запустить приложение. Для этого необходимо нажать функциональную клавишу F9 или кнопку панели инструментов:



16. Заполняем поля таблицы соответствующими по смыслу значениями. Закрываем запущенное приложение, и перед нами вновь окна DELPHI в режиме конструктора. Новые данные, которые вы ввели в таблицу, пока не будут отображаться. Для того, чтобы они появились, необходимо в свойстве Active компонента ADOTable выбрать сначала значение False, а затем True.
17. Для удобства работы с записями таблицы, необходимо установить на форме визуальный компонент DBNavigator с вкладки DataControls.



18. Для того чтобы компонент Навигатор взаимодействовал с таблицей, необходимо осуществить привязку к источнику данных, как показано на рисунке:

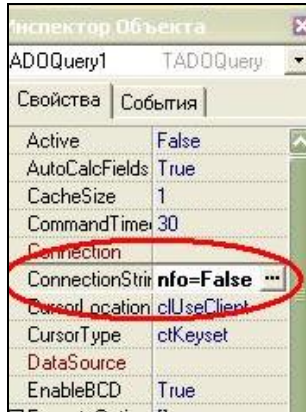


Разработка запросов

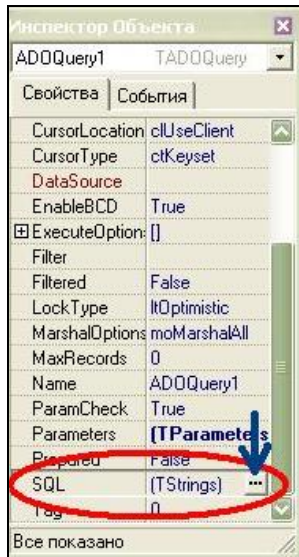
1. С вкладки ADO выбираем невидимый компонент ADOQuery (Запрос)



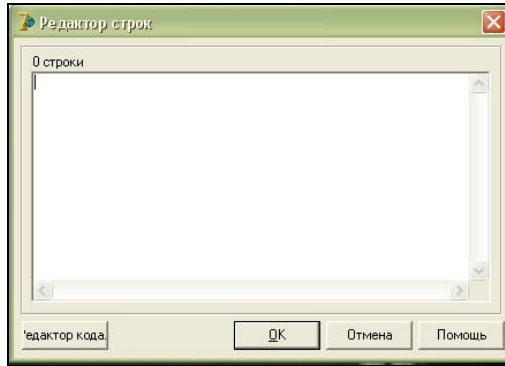
2. На форме расположим еще компоненты Data Source и DBGrid. Подключим компонент ADOQuery к базе данных, выполнив для его свойства ConnectionString те же операции, что и для ADOTable. В этом случае важно не указывать название таблицы.



3. Основным свойством для формирования текста запроса, является свойство SQL компонента ADOQuery



При нажатии мышью на кнопку, указанную стрелкой, появляется окно «Создания запроса».

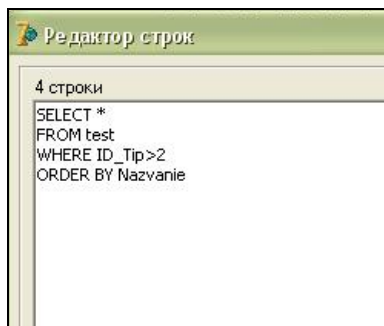


Здесь вводится текст запроса. Переход на новую строку осуществляется с помощью клавиши <Enter>.

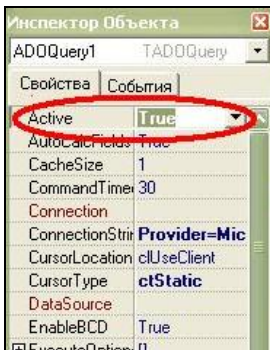
4. Для того чтобы сформировать запрос, необходимо определить, что именно мы хотим получить в результате выполнения запроса. Общая структура простейшего запроса выглядит так:

```
SELECT <список отображаемых полей> или *( все поля)
FROM   <список таблиц>
WHERE  <условие выбора записей>
ORDER BY <список полей для сортировки>
```

Пример. Придерживаясь данной структуры, создаем свой запрос, который будет осуществлять выбор записей по полю ID_Tip, значения которого удовлетворяют условию >2 . Результат должен быть отсортирован по полю Nazvanie в алфавитном порядке:



После ввода текста запроса нажимаем кнопку «ОК». Теперь активируем запрос, для чего в свойстве Active компонента ADOQuery укажем значение True.



В программе для этой цели можно использовать также метод ADOQuery1.Open.

Если исходные данные таблицы выглядят так:

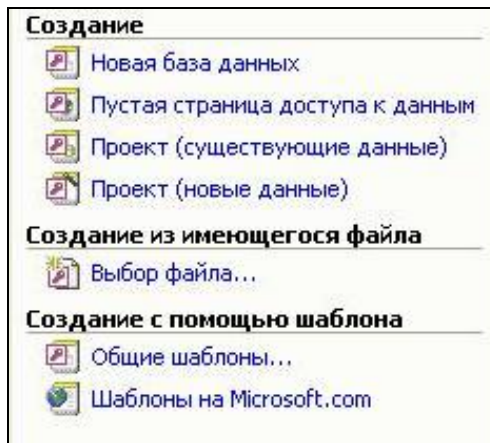
ID_Tip	Nazvanie
1	name1
2	name2
3	name3
4	name4

то после выполнения запроса результат будет следующий:

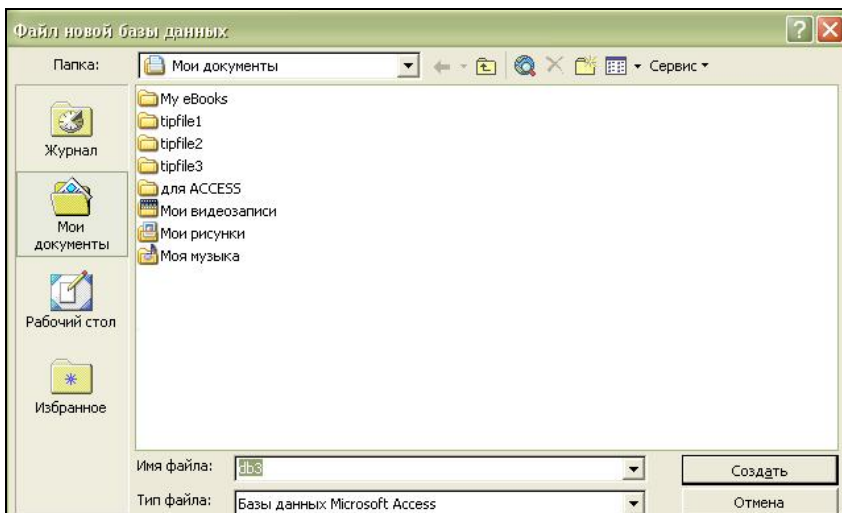
ID_Tip	Nazvanie
3	name3
4	name4

СОЗДАНИЕ ТАБЛИЦ В СУБД MICROSOFT ACCESS 2000

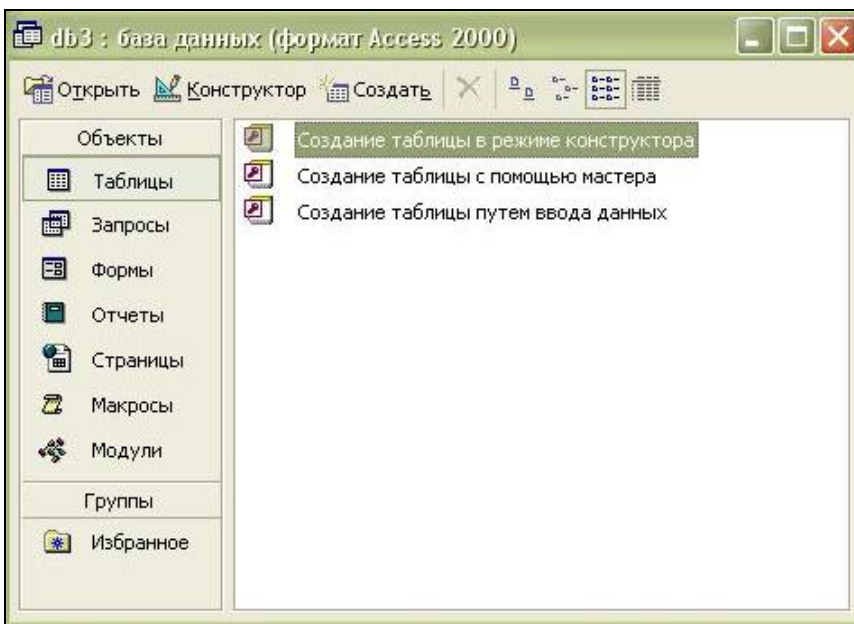
1. Запускаем СУБД: Пуск – Все программы – Microsoft Access.
2. В окне приложения выбираем команду Файл – Создать.
3. Появляется панель с перечнем команд:



4. Выбираем команду «Создание – Новая база данных».
5. Открывается диалоговое окно «Файл новой базы данных». В данном окне необходимо написать имя создаваемой базы данных и выбрать местоположение, где эта база будет храниться. В нашем примере имя новой базы данных – «db3», сохраним ее в папке «Мои документы».

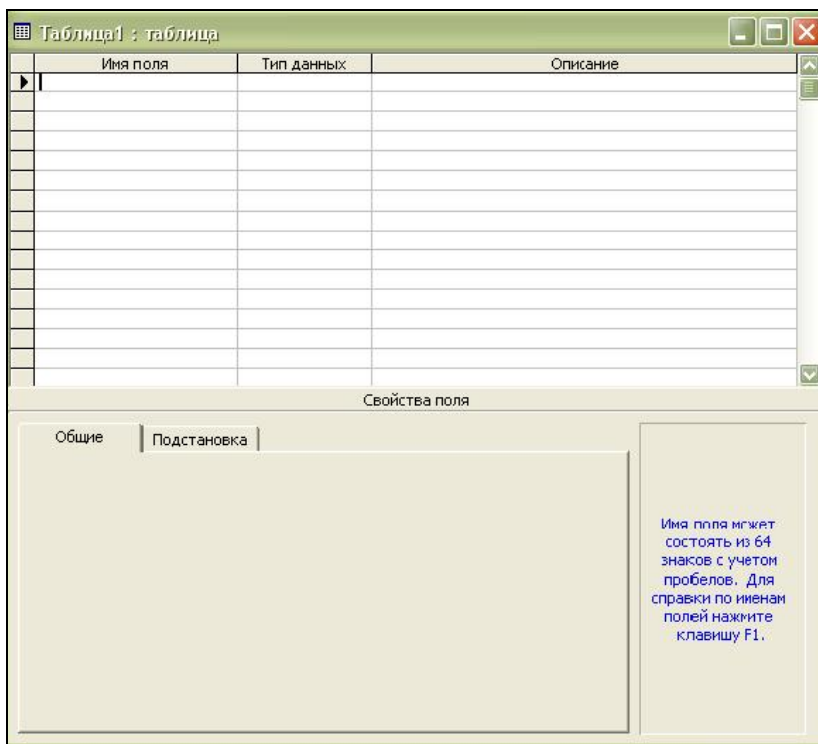


6. При нажатии на кнопку «Создать» появляется окно базы данных MS Access, которое выглядит следующим образом:



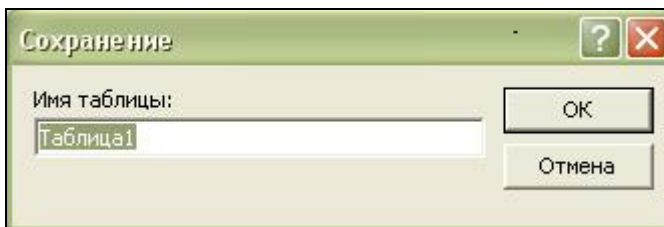
Слева в данном окне можно выбрать необходимую вкладку: Таблицы, Запросы, Формы, Отчеты, Страницы, Макросы, Модули.

7. Так как нам необходимо создать таблицу, то необходимо перейти на вкладку Таблицы и выбрать команду «Создание таблицы в режиме конструктора». Откроется окно структуры таблицы:

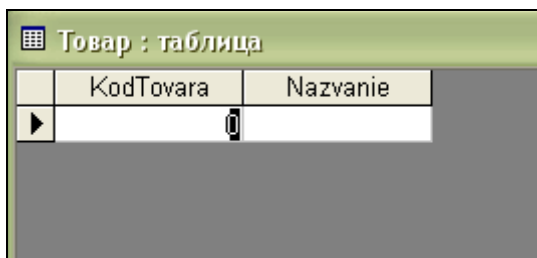


В данном окне мы, формируем список полей таблицы, указывая их тип и все необходимые дополнительные свойства.

8. Создадим таблицу «Товар», которая будет содержать список товаров с их кодами и названиями. Для этого формируем два поля: Код товара и Название.




При желании можно посмотреть данные в таблице, это осуществляется при двойном щелчке по имени таблицы, в результате появляется окно:



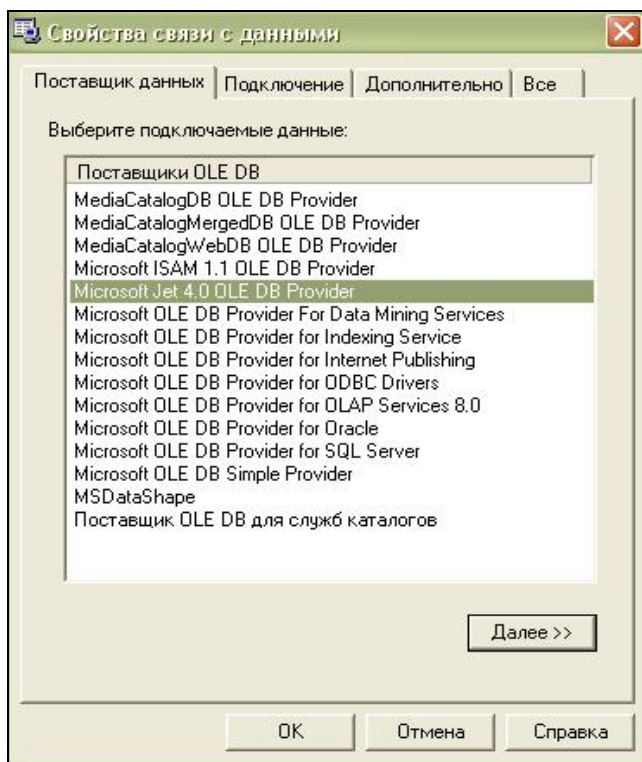
На этом создание таблиц в СУБД MS Access закончено.

Подключение таблиц MS Access к компонентам Delphi

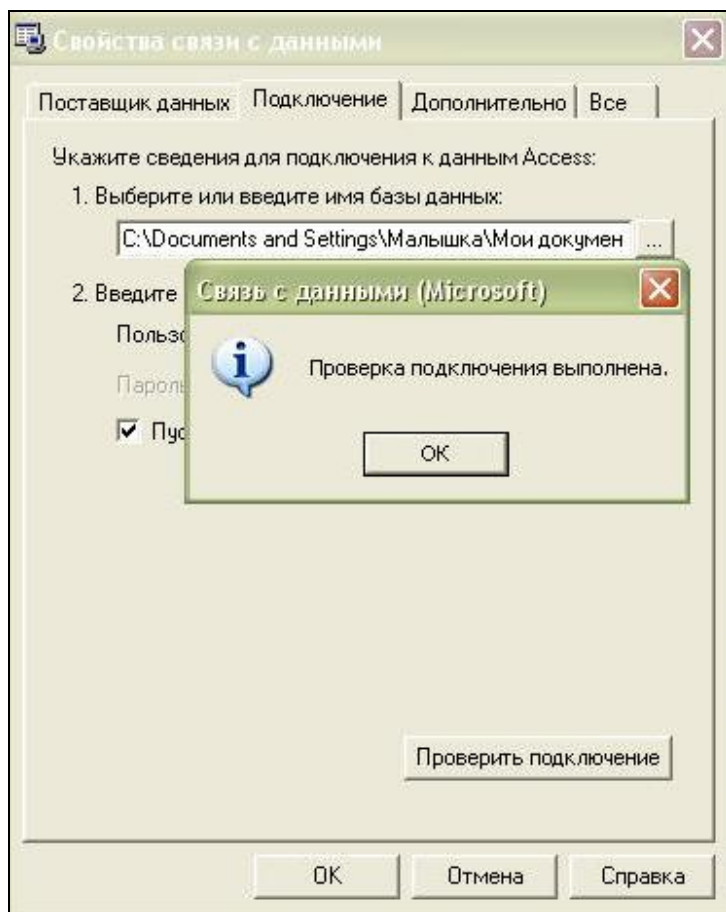
1. Заходим в среду Delphi: Пуск – Программы – Borland Delphi 7 – Delphi 7.
2. Для подключения таблиц MS Access будем использовать механизм доступа ADO.
3. С вкладки ADO палитры компонентов выбираем компонент ADOTable 
4. В свойствах компонента ADOTable выбираем свойство ConnectionString и нажимаем на кнопку с 3 точками.
5. Появится диалоговое окно, в котором необходимо нажать на кнопку Build (т.е. использовать строку соединения).



6. Появится диалоговое окно «Свойства связи с данными». Для связи Access с Delphi необходимо выбрать из предложенного списка Поставщиков OLE DB пункт Microsoft Jet 4.0 OLE DB Provider и нажать на кнопку «Далее».



7. В появившемся диалогом окне «Свойства связи с данными» в строке «Выберите или введите имя базы данных» необходимо указать путь к созданной базе данных. Желательно не прописывать путь к базе вручную, а использовать кнопку с тремя точками. После того, как вы указали путь к базе данных, вам необходимо нажать на кнопку «Проверить подключение».



8. Откроется окно с надписью «Проверка подключения выполнена», в котором следует нажать кнопку «ОК». В

появившемся затем окне в строке Use Connection String вы увидите путь к вашей базе данных. Если же данная строка пуста, то необходимо вновь нажать на кнопку Build и возобновить процедуру. Если вы все сделали правильно, то в результате строка соединения должна выглядеть примерно так:



Нажимаем на кнопку ОК и возвращаемся в интегрированную среду разработки Delphi.

9. Закрываем все окна, нажимая “OK”. В результате в свойстве компонента ADOTable в строке ConnectionString появляется путь к вашей базе данных.
10. В свойстве TableName компонента ADOTable нужно указать имя таблицы. В нашем случае имя таблицы - «Товар». В свойстве Active компонента ADOTable указываем значение True, то есть открываем таблицу. Далее все действия аналогичны тем, которые выполнялись при подключении таблиц MS SQL Server к соответствующим компонентам Delphi.

Связь таблиц

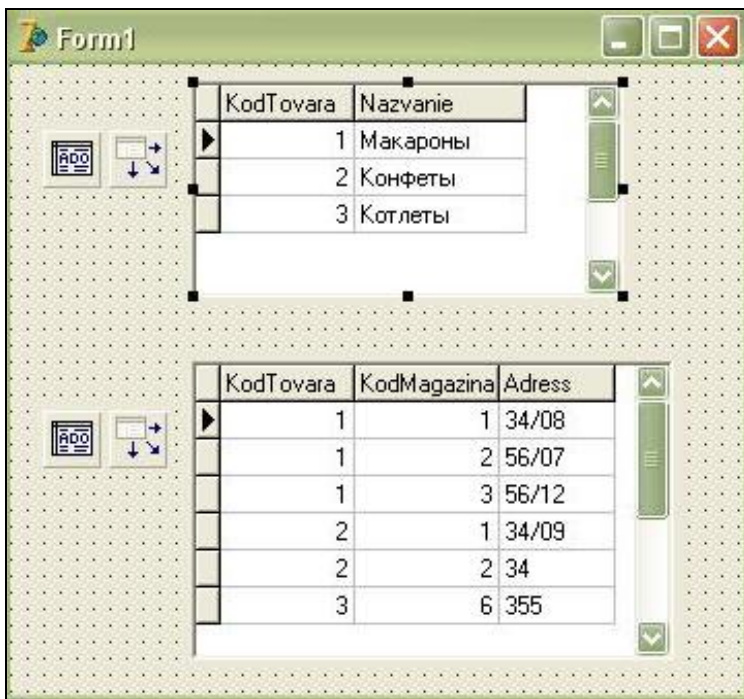
Две таблицы в Delphi можно связать с помощью свойств MasterSource/MasterFields. Таблицы связываются друг с другом по ключевому или по индексированному полю.

Создадим в СУБД MS Access вторую таблицу «Магазин», которая будет содержать поля Код товара, Код

магазина и Адрес. Ключевыми полями здесь будут являться Код товара и Код магазина (составной ключ).

Организуем доступ к данной таблице в Delphi через компоненты ADOTable, DataSource, DBGrid как было показано выше.

Заполним таблицы «Товар», «Магазин»:

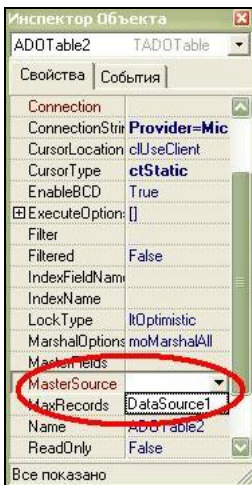


Перед тем как организовывать связь, необходимо определить, какая из таблиц у вас будет являться главной, а какая - подчиненной. Тип связи будет «Один ко многим», то есть одной записи в главной таблицы будет соответствовать несколько записей из подчиненной таблицы, и каждой записи в подчиненной таблицы будет соответствовать только одна запись в главной таблице.

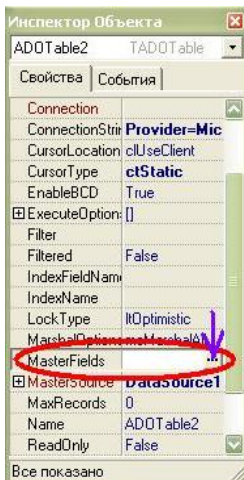
В нашем примере главной таблицей будет являться таблица «Товар», а подчиненной - таблица «Магазин».

Организация связи в Delphi

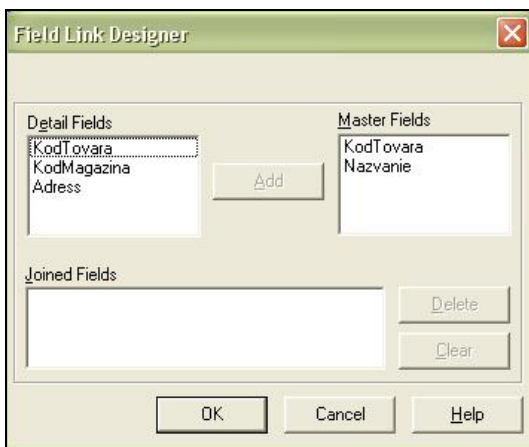
1. Выделяем подчиненную таблицу «Магазин» (например, щелкнув мышью на компоненте ADOTable2).
2. В Инспекторе Объектов для данного компонента находим свойство MasterSource и выбираем источник данных для первой таблицы, то есть источник данных для главной таблицы «Товар» - DataSource1.



3. Затем в Инспекторе Объектов находим свойство MasterFields. Нажимаем на кнопку с 3 точками справа от этого свойства:

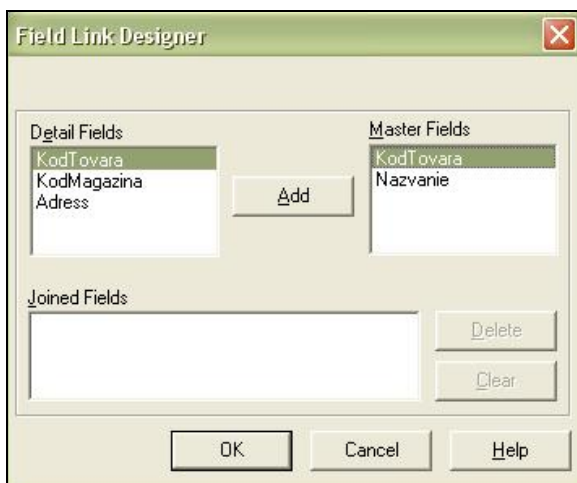


4. После нажатия появляется диалоговое окно Field Link Designer.

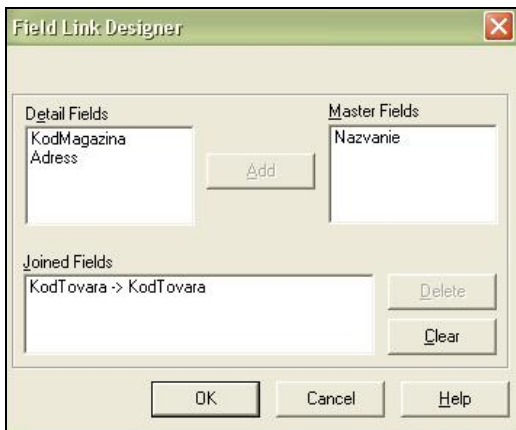


В поле Detail Fields перечислены поля подчиненной таблицы «Магазин», а в поле Master Fields – поля главной таблицы «Товар».

5. Свяжем таблицы по ключевому полю KodTovara. Для этого необходимо выделить поле KodTovara в списках Detail Fields и Master Fields:



Нажимаем на кнопку «Add» - в поле Joined Fields появляется организованная связь.



6. Завершаем процесс создания связи с помощью кнопки «OK». Теперь можно увидеть результат связывания таблиц в работающем приложении. Например, при выделении в главной таблице «Товар» записи (строки) с кодом товара под номером один, в подчиненной таблице будут видны только те записи, которые соответствуют коду товара равному единице.



При выделении в главной таблице «Товар» записи с кодом товара под номером два, в подчиненной таблице «Магазин» появляются только те записи, которые соответствуют коду товара равному двум и так далее.

Организация такой связи значительно упрощает процесс восприятия данных, позволяет систематизировать данные таблиц.

В результате мы можем легко узнать, какие магазины заказали данный вид товара.

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

1. В.В.Фаронов. Delphi. Программирование на языке высокого уровня: учебник для ВУЗов // СПб.:Питер, 2006. - 640 с.: ил. - ISBN 5-8046-0008-7.
2. А.Хомоненко, В.Гофман, Е.Мещеряков, В.Никифоров. Delphi 7: Визуальная разработка приложений. Свойства, методы, классы, компоненты. Технология создания информационных систем. Локальные и удаленные базы данных. Публикация баз данных в Интернете // СПб.: БХВ-Петербург, 2006. - 1216 с.: ил. – ISBN 5-94157-267-0.
3. А.В.Сорокин. Delphi. Разработка баз данных // СПб.:Питер, 2005. - 477 с.: ил. - ISBN 5-469-00927-0.
4. С.И.Бобровский. Delphi 7. Учебный курс // Спб.: Питер, 2005. – 736 с.: ил. – ISBN 5-8046-0086-9.
5. А.Г.Исавнин, А.Б.Ершов. Основы алгоритмизации. Часть 3. Delphi. Работа с базами данных. Методические указания // Набережные Челны: Изд-во КамПИ, 2000 г. – 35 с.
6. А.Г.Исавнин, И.В.Макарова, Д.М.Лысанов. Язык структурированных запросов. Учебное пособие // Набережные Челны: Изд-во КамПИ, 2004 г. – 34 с.
7. А.Г.Исавнин, Д.М.Лысанов, В.С.Фрикк. Создание отчетов с помощью RAVE – компонентов Delphi. Учебное пособие // Набережные Челны: Изд-во ИНЭКА, 2006 г. – 39 с.