

## Лабораторная работа № 7

### Использование утилиты ssms для работы с сервером Microsoft sql Server 2008

**Цель работы:** получить навыки работы с утилитой SSMS, обеспечивающей доступ к базам данных сервера Microsoft SQL Server.

#### Теоретические сведения

Microsoft SQL Server, или SQL Server, - одна из наиболее мощных СУБД, имеющая клиент-серверную архитектуру. Особенностью СУБД является возможность ее работы только в операционных системах Microsoft Windows, при этом клиентская часть может взаимодействовать с СУБД, работая под управлением Microsoft Windows и других операционных систем.

В составе системы имеются средства создания баз данных, работы с информацией баз данных, переноса данных из других систем и в другие системы, резервного копирования и восстановления данных, реляционная подсистема для анализа, оптимизации и выполнения запросов клиентов, система безопасности для управления правами доступа к объектам базы данных и т.п.

Для правильного проектирования баз данных необходимо знание типов данных, которые могут использоваться для полей таблиц в базе. В табл.1 представлены типы данных в Microsoft SQL Server с разбивкой на группы по видам. В табл.2 приведены некоторые характеристики СУБД.

SQL Server имеет входной язык Transact-SQL (T-SQL), в котором помимо базовых SQL-операторов предусмотрены программные конструкции - параметры, переменные и логические структуры (IF, WHILE и т.д.). В своей основе язык T-SQL совместим со стандартом ANSI SQL-92.

**Таблица 1**

**Типы данных субд Microsoft sql Server**

Тип	Описание
Двоичные данные	
binary [ ( n ) ]	Максимальная длина 8 000 байт (n)
varbinary [ ( n ) ]	Данные переменной длины, максимальная длина 8 000 байт (n)
image	Максимальная длина 2 147 483 647 байт
bit	Тип данных, который принимает значения 1 или 0
Символьные данные	
char [(n)]	Максимальная длина 8 000 символов (n)
varchar [(n)]	Тип переменной длины, максимальная длина 8 000 символов (n)
text	Максимальная длина 1 073 741 823 символов

Тип	Описание
Символьные данные в кодировке Unicode	
nchar (n)	Максимальная длина 4 000 символов (n)
nvarchar (n)	Тип переменной длины в кодировке Unicode, максимальная длина 4 000 символов (n)
ntext	Максимальная длина 1 073 741 823 символов
Числовые целые данные	
bigint	Диапазон от -922 337 203 685 4775808 до 922 337 203 685 4775807
int	Диапазон от -2 147 483 648 до 2 147 483 647
smallint	Диапазон от -32 768 до 32 767
tinyint	Диапазон от 0 до 255
Числовые данные с дробной частью числа	
decimal[(p[, s])]	Диапазон от $-10^{38}-1$ до $10^{38}-1$ с заданием фиксированного количества знаков (p - всего, s - в дробной части), максимальное общее количество знаков 38
numeric	То же, что и decimal
float [ ( n ) ]	Диапазон от $+2.29 \cdot 10^{-308}$ до $+1.79 \cdot 10^{308}$
real	Числа с 7-значной точностью в диапазоне от $+1.18 \cdot 10^{-38}$ до $+3.40 \cdot 10^{38}$
Тип дата и время	
datetime	Диапазон от 1.01.1753 до 31.12.9999 с точностью 3.33 мс

*Окончание*

Тип	Описание
smalldatetime	Диапазон от 1.01.1900 до 6.06.2079 с точностью 1 мин.
Денежный тип	
money	Диапазон от -922 337 203 685 477.5808 до +922 337 203 685 477.5807
smallmoney	Диапазон от -214 748.3648 до +214 748.3647
Данные специальных типов	
timestamp	Счетчик, автоматически увеличивающийся, имеющий уникальное значение для базы данных (тип binary(8) или varbinary(8))
uniqueidentifier	Тип, который содержит уникальный идентификационный номер (GUID), сохраняемый как 16-битная двоичная строка
sql_variant	Тип, который сохраняет значения различных типов, кроме text, ntext, timestamp и sql_variant
sysname	Тип - синоним nvarchar, используется для ссылок на имена объектов базы данных

## Некоторые характеристики Microsoft sql Server

Характеристика	Максимальное значение
Размер базы данных	1 048 516 Тбайт
Количество объектов в базе данных	2 147 483 647
Количество экземпляров сервера на одном компьютере	16
Количество баз данных в одном экземпляре сервера	32767
Количество файлов в базе данных	32767
Количество таблиц в базе данных	Ограничено количеством объектов в базе
Количество полей в таблице базы	1024
Размер файла данных	32 Тбайт
Длина идентификаторов	128 символов
Уровень вложенных запросов	32
Количество полей в одном индексе	16
Количество таблиц в одном запросе	256

Реализация логики приложения в SQL Server возможна тремя способами: посредством процедур на T-SQL и хранимых запросов, посредством хранимых процедур, которые вызываются из прикладных программ, и посредством триггеров, которые вызывает SQL Server при выполнении определенных действий с базой данных.

### Утилита ssms

Утилита SSMS, называемая также средой SQL Server Management Studio, предоставляет пользовательский интерфейс для работы с сервером SQL Server. Она имеет набор средств, аналогичный имеющемуся в Microsoft Visual Studio, и позволяет разработчикам и администраторам баз данных создавать базы данных и управлять СУБД SQL Server как с помощью команд пользовательского интерфейса, так и с помощью операторов языка T-SQL.

Утилита SSMS посылает операторы языка T-SQL серверу или использует специальные объекты управления сервером, а также инспектирует сервер и отображает его объекты и конфигурацию в понятном пользователю представлении.

**Организация пользовательского интерфейса.** Доступ к функциям утилиты SSMS осуществляется с помощью панелей (окон), которые можно открыть командой главного меню View (Вид), кнопками стандартной панели инструментов или “горячими” клавишами, ассоциированными с командами (рис.1):

**Object Explorer (F8)** - панель **Обозревателя объектов** используется для создания и администрирования объектов базы данных;

**Object Explorer Details (F7)** - панель **Сводки** содержит основную информацию о выделенном объекте, а также используется для отображения некоторых отчетов;

**Registered Servers (Ctrl+Alt+G)** - панель **Зарегистрированных серверов** используется для управления подключениями к нескольким ядрам сервера, а также для регистрации ядра серверов баз данных, служб анализа, отчетности и интеграции;

**Template Explorer (Ctrl+Alt+T)** - панель **Обозревателя шаблонов** используется для управления шаблонами программ на языке T-SQL;

**Properties (F4)** - панель **Свойств** отображает свойства выделенного объекта;

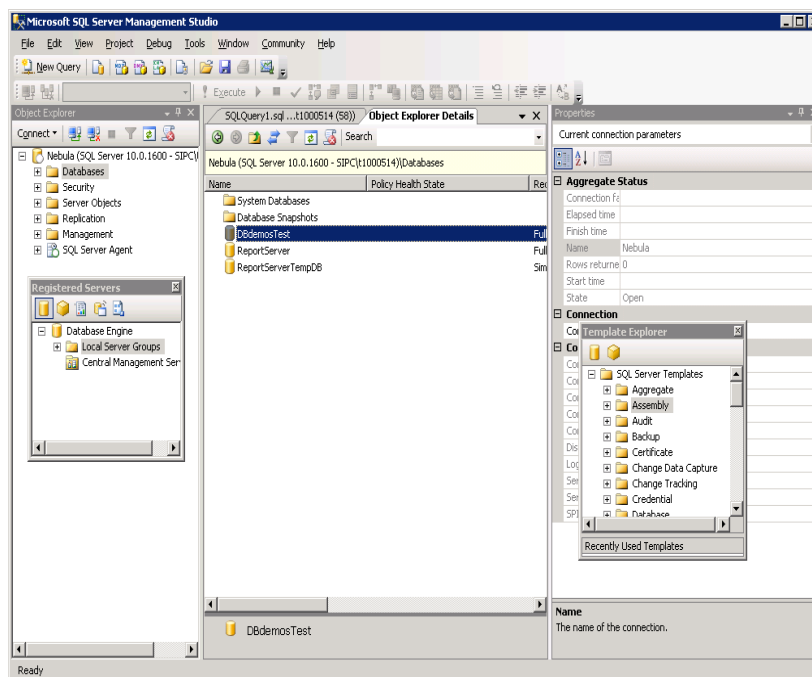


Рис. 1. Главное меню и некоторые панели утилиты SSMS

**Query Editor** - панель **Редактора запросов** позволяет создавать, изменять и выполнять пакеты операторов на языке T-SQL. Запросом называют один оператор языка T-SQL, а пакетом - их набор. Панель открывается командой главного меню File | New (Файл | Создать), кнопкой New Query (Создать запрос) панели инструментов, а также при создании запроса или сценария для объекта на панели **Object Explorer**.

В утилите SSMS основным средством выбора действий и просмотра свойств является контекстное меню, которое связано с большинством типов объектов и содержит команды создания новых объектов и выполнения операций с объектами.

**Управление размещением окон.** Любое окно можно сделать прикрепленным (закрепляемым) или плавающим, включить как закладку в другое окно, а также скрывать за одной из границ главного окна утилиты. Необходимый режим можно задать с помощью щелчка правой кнопки мыши на заголовке окна (из контекстного меню), выбора значка со стрелкой вниз на крайней правой границе прикрепленного окна или с помощью команды главного меню Window. Также можно автоматически изменить режим отображения окна, перетаскивая его за заголовок в соответствующее место экрана. Доступны следующие режимы для управления размещением окна:

- установка режима окна floating (плавающее) открепляет его от границы главного окна утилиты. Далее оно ведет себя как обычное немодальное диалоговое окно;
- установка режима tabbed (режим вкладки) немедленно перемещает его в разделенный на вкладки документ в центре главного окна утилиты в качестве одной из вкладок. Порядок вкладок можно изменить с помощью перетаскивания их указателем мыши. При перетаскивании вкладки в положение отдельно от документа автоматически создается новый документ. В любом положении (в центре, справа, слева, вверху и внизу) может находиться несколько разделенных на вкладки инструментов и документов;
- рабочая область документа может хранить больше вкладок, чем может на ней физически поместиться. Существует два способа просмотра скрытых вкладок. Наиболее очевидным из них является использование полосы горизонтальной прокрутки. Эффективнее всего развернуть список документов с помощью стрелки Active File (Активные файлы) в верхнем правом углу рабочей области документа;
- при перемещении окна в режиме dockable (прикрепленное или закрепляемое) утилита отображает несколько синих индикаторов прикрепления (рис.2). Если оставить окно около одного из этих индикаторов, оно будет прикреплено к соответствующей границе. Если оставить окно около индикатора в центре окна, то оно превратится в документ, разделенный на вкладки;
- в режиме автоматического сокрытия окно становится невидимым до тех пор, пока пользователь не щелкнет на соответствующей вкладке, расположенной у левой или правой границы главного окна утилиты. Для настройки этого режима служит пункт главного меню Window | Auto Hide или щелчок на значке булавки в заголовке окна. Когда булавка находится в вертикальном положении, окно остается открытым, в противном случае оно автоматически скрывается. Автоматически скрываемое окно нужно снова открыть, прежде чем можно будет изменить его режим на плавающий или режим вкладки (см. рис.2).

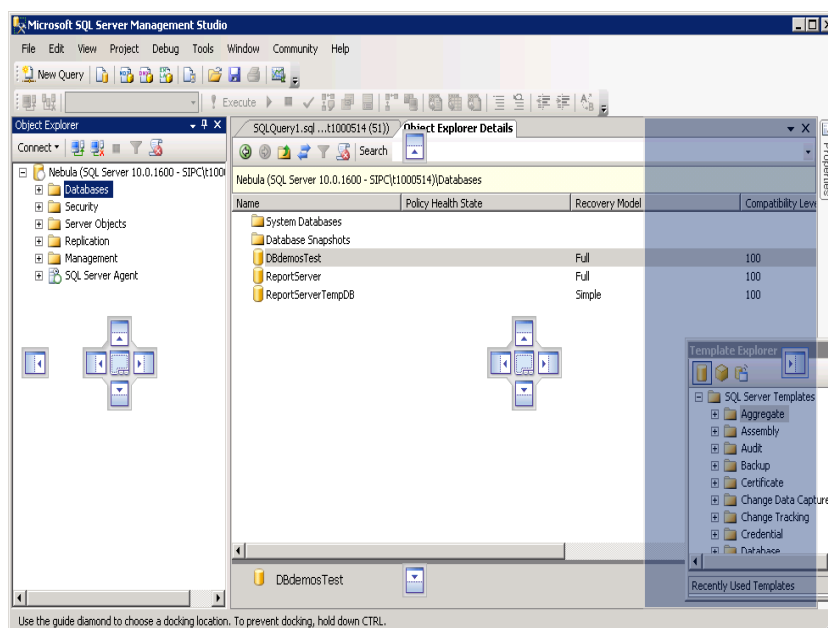


Рис.2. Управление размещением окна с помощью маркеров

Изменить режим отображения окон на принятый по умолчанию (т.е. будут открыты только окна Object Explorer, Tabbed Documents и Properties) позволяет команда Window | Reset Window Layout. Эта команда не способна восстановить какой-либо измененный пользователем режим.

Для переключения между открытыми окнами используют комбинацию клавиш Ctrl+Tab.

Для сокрытия всех прикрепленных окон и сохранения на экране только открытых в центре документов с вкладками используют команду Window | Auto Hide All.

**Панель обозревателя объектов.** Окно Object Explorer отображает все объекты сервера. На верхнем уровне дерева перечисляются все подключенные серверы. Можно подключиться к любому серверу независимо от того, содержится ли он в списке зарегистрированных серверов, который находится в окне Registered Servers. Цвет значка сервера отражает его текущее состояние - запущен он или остановлен.

Подобно Проводнику Windows, окно Object Explorer (рис.3) является иерархическим представлением объектов, доступных в подключенных серверах. Все дерево состоит из корней и узлов. Например, в дереве папок Проводника Windows узел Рабочий стол является корневым, а все остальные папки и устройства раскрываются как его отдельные подузлы.

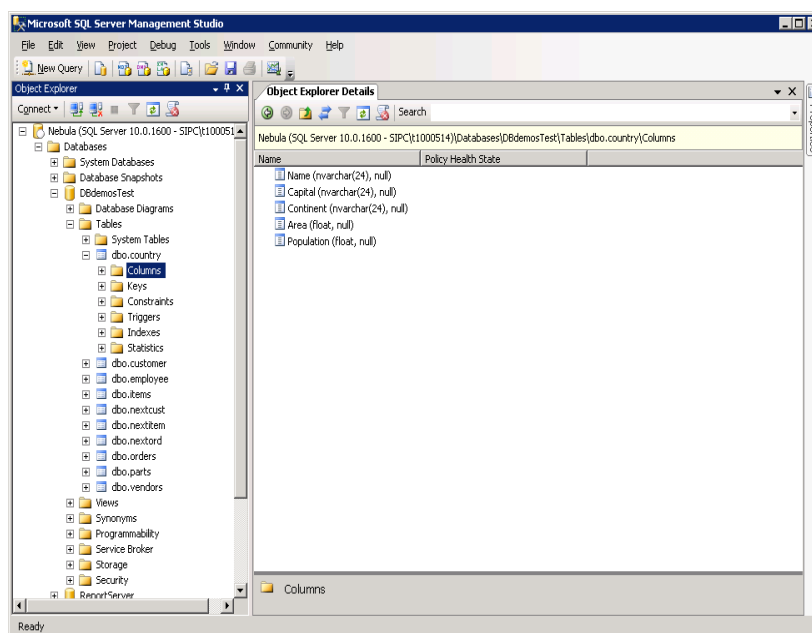


Рис.3. Структура дерева на панели Обозревателя объектов обеспечивает доступ к различным компонентам сервера и средствам управления и разработки

Среди узлов сервера можно встретить базы данных, средства защиты, серверные объекты, репликацию, управление. Большая часть структуры этого дерева фиксирована, но в процессе работы в нее автоматически добавляются объекты, создаваемые на сервере.

Узел Databases содержит все базы данных, находящиеся на сервере. Если щелкнуть правой кнопкой мыши на любой из этих баз данных, откроется контекстное меню, содержащее основные операции, выполняемые над базой данных. Узел каждой базы данных содержит связанные с ней подузлы (рис.4), представляющие следующие объекты базы данных:

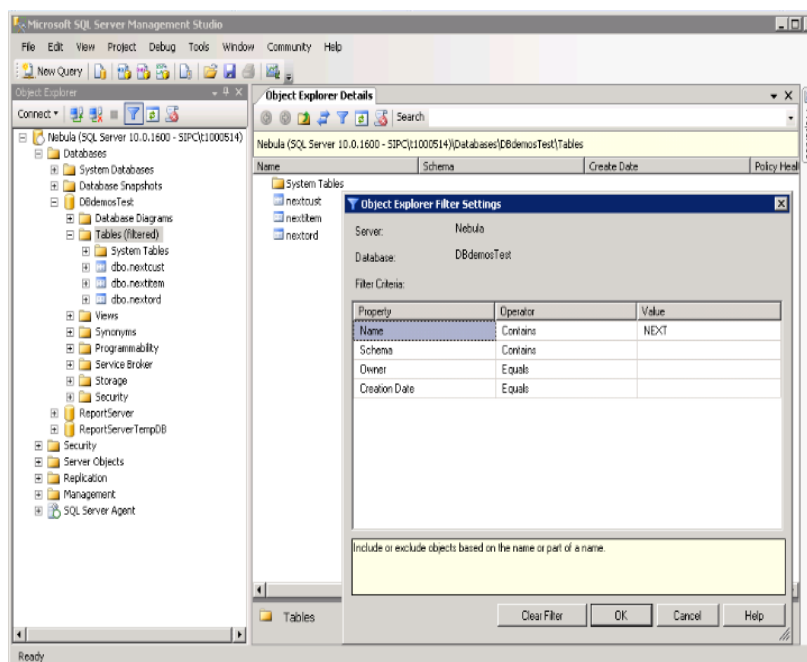


Рис.4. Диалоговое окно Filter Settings (Настройка фильтра) используется для ограничения числа объектов, отображаемых на панели Обзорателя объектов

- диаграммы баз данных (Database Diagrams). Диаграммы отображают в графическом представлении отдельные таблицы базы и связи между ними. Одна база данных может иметь несколько диаграмм, при этом в каждой из диаграмм не обязательно отображаются все таблицы. Такой подход облегчает организацию крупных баз данных в модульные диаграммы;
- таблицы (Tables). Этот подузел используется для создания и модификации структур таблиц, а также для работы с их индексами, ограничениями и статистикой. Здесь могут создаваться и редактироваться триггеры и хранимые процедуры, отвечающие за операции модификации данных (вставку, обновление и удаление);
- представления (Views). В этом подузле можно создавать и редактировать операторы создания представлений, а также просматривать результаты их работы;
- синонимы (Synonyms). Здесь содержатся альтернативные имена объектов базы данных;
- программирование (Programmability). В этом обширном разделе содержится большая часть объектов разработки, хранимых процедур, функций, триггеров базы данных, сборок, типов данных, правил и значений по умолчанию;
- брокер служб (Service Broker). Этот подузел используется для просмотра содержимого асинхронных очередей брокера служб;
- хранение (Storage). Этот подузел используется для управления нестандартными хранилищами и содержит такие операции, как полнотекстовый поиск и разбиение таблиц на разделы;

- безопасность (Security). Этот подузел используется для управления средствами защиты базы данных.

Так как утилита SSMS связывается с сервером в качестве клиента, то процессы, соответствующие утилите и серверу, не всегда синхронизированы. Поэтому изменения, выполняемые на сервере, могут отображаться в утилите с запаздыванием. Для выполнения синхронизации обычно требуется обновить содержимое окна нажатием кнопки Refresh (Обновить).

Базы данных могут содержать большое количество объектов. Для облегчения навигации между ними предусмотрена фильтрация фрагмента дерева, содержащего пользовательские объекты, такие как таблицы и представления. Кнопка фильтрации Filter (Фильтр) с пиктограммой в виде воронки находится на панели инструментов окна Object Explorer. Эта кнопка активизирована только в том случае, когда выделен узел, соответствующий пользовательскому объекту. Например, для фильтрации таблиц нужно выделить соответствующий узел дерева, щелкнуть на кнопке Filter и настроить фильтр, задав критерии (условия) фильтрации (см. рис.4).

Диалоговое окно Filter Settings позволяет фильтровать объекты по имени, схеме, владельцу и дате создания. Для того чтобы снять фильтр, можно воспользоваться контекстным меню или открыть диалоговое окно Filter Settings и щелкнуть на кнопке Clear Filter. К сожалению, фильтр для каждого из параметров может задаваться только одним значением и булевы операции здесь недопустимы.

**Использование редактора запросов.** Редактор запросов (Query Editor) активизируется командой главного меню File | New (Файл | Создать) или кнопкой New Query (Создать запрос) панели инструментов и позволяет пользователю работать с несколькими окнами, открытыми в области документов с вкладками (рис.5). В заголовке окна отображаются имя зарегистрировавшегося пользователя и имя сервера, а также имя и расширение файла, в котором будет сохранен или уже хранится запрос или пакет.

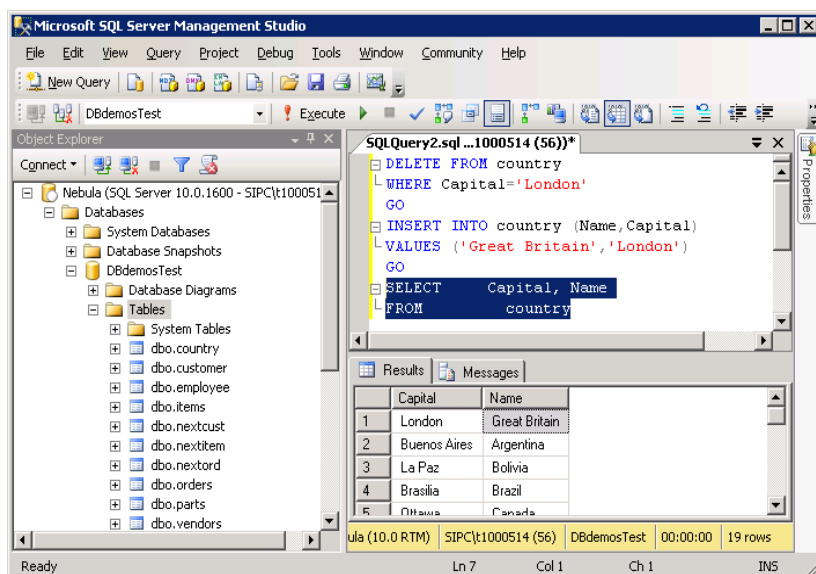


Рис.5. Редактор запросов с пакетом, в котором выделен один оператор



Запрос - это один оператор языка T-SQL, а пакет - их набор. Вся последовательность операторов пакета отправляется серверу из клиентских приложений как одно целое.

SQL Server рассматривает весь пакет как рабочую единицу. Наличие ошибки хотя бы в одном операторе приведет к невозможности выполнения всего пакета.

В содержащем SQL-операторы файле сценария, имеющем расширение .sql, и в окне редактора запросов могут находиться несколько пакетов. В этом случае все пакеты разделяют ключевые слова терминаторов. По умолчанию этим ключевым словом является GO, и оно должно быть единственным в строке. Все другие символы (даже комментарии) нейтрализуют действие разделителя пакета.

Будучи средством разработки редактор запросов позволяет выполнять пакеты T-SQL, состоящие из множества операторов этого языка. Пакет передается серверу для обработки командой меню Query | Execute, щелчком на кнопке Execute панели инструментов, нажатием клавиши F5 или комбинации клавиш Ctrl+E.

Так как пакеты операторов, как правило, довольно длинные, часто предпочтительнее для тестирования выполнять отдельные операторы пакета или его фрагменты. Если не выделен какой-либо фрагмент текста, то будет выполнен весь пакет, если выделен, то будет выполнен только выделенный фрагмент.

Следует отметить, что команда меню Query | Parse и соответствующая кнопка панели инструментов позволяют проверить только правильность кода SQL. Эта функция не проверяет имена объектов (таблиц, столбцов, хранимых процедур и т.п.), а всего лишь проверяет синтаксис инструкций SQL, но в то же время SQL Server позволяет создавать в пакетах объекты и затем ссылаться на них в этих пакетах.

Пакет T-SQL будет выполняться в контексте текущей базы данных. Текущая база данных отображается в поле выпадающего списка панели инструментов и при желании может быть изменена.

Результаты выполнения запроса отображаются в нижней области, при этом формат может быть как табличным, так и текстовым - между ними можно переключаться с помощью комбинаций клавиш Ctrl+D и Ctrl+T соответственно или кнопок Results to Grid и Results to Table панели инструментов. Новый формат будет применен при выполнении следующего пакета.

При работе с кодом T-SQL в редакторе запросов можно получить справку по выделенному ключевому слову, нажав комбинацию клавиш Shift+F1. В качестве альтернативы в утилите SSMS можно открыть окно динамической справки, при этом работа пользователя будет отслеживаться, а в этом окне отображаться справка по соответствующей теме.

Одной из самых существенных функций редактора запросов является возможность просматривать план выполнения запроса в графическом представлении (рис.6). Планы выполнения важны потому, что SQL является декларативным, т.е. описательным языком - он не указывает оптимизатору, как именно извлекать данные, а лишь описывает, какие данные ему нужны. Хотя некоторого улучшения производительности можно добиться за счет правильного моделирования

оператора, основная настройка выполняется правильной установкой индексов - именно они влияют на то, как оптимизатор запросов скомпилирует запрос. План выполнения запроса указывает серверу, как оптимизировать запрос, как использовать существующие индексы, как запросить данные из других источников и создать соединения. Редактор запросов может отображать примерный план перед выполнением запроса (кнопка Display Estimated Execution Plan или аналогичная команда меню Query) или действительный план уже после его выполнения (кнопка Include Actual Execution Plan или команда меню Query).

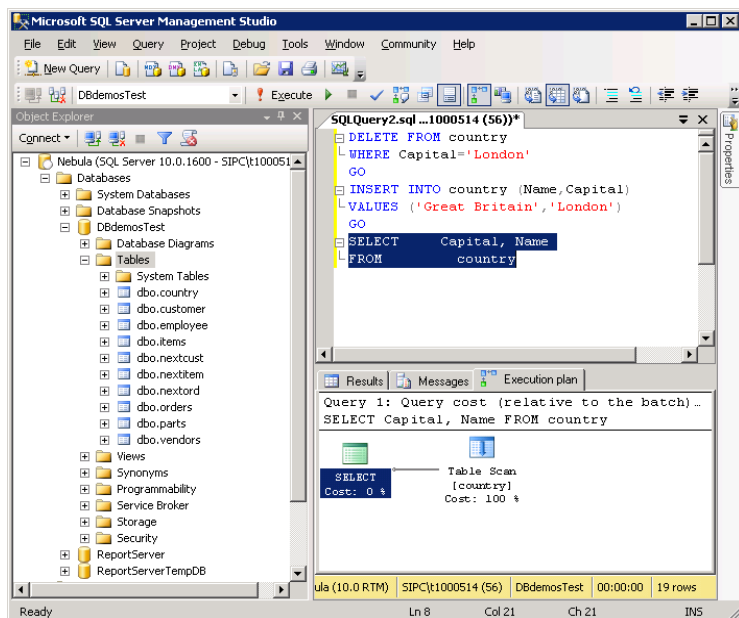


Рис.6. Графическое отображение плана выполнения запроса в редакторе запросов

**Конструктор запросов.** Конструктор запросов (Query Designer) - удобное средство извлечения и модификации данных, хотя его нельзя назвать простейшим инструментом утилиты. Открыть его можно несколькими способами.

Из редактора запросов (Query Editor) конструктор можно открыть с помощью команды меню Query | Design Query in Editor или одноименной кнопки панели инструментов. При этом конструктор запросов предлагает выбрать из текущей базы данных таблицы, для которых будет формироваться запрос (рис.7). Нужная таблица указывается в списке и выбирается нажатием кнопки Add. Графическое представление выбранных таблиц автоматически появляется в окне конструктора.

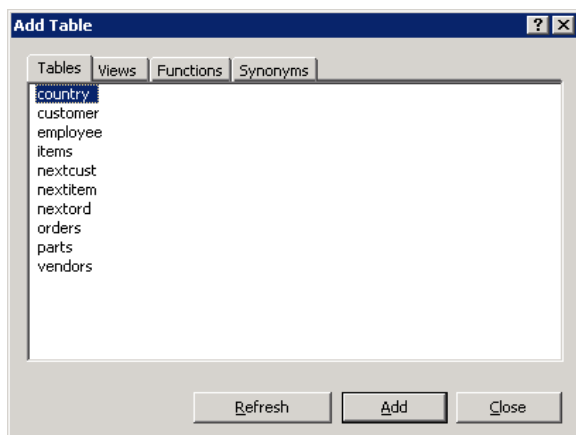


Рис.7. Окно выбора таблиц для формируемого запроса

Другой способ активизации конструктора запросов заключается в выделении любого запроса в редакторе запросов, вызове контекстного меню нажатием правой кнопки мыши и выборе команды Design Query in Editor.

В отличие от других инструментов работы с запросами, переключающихся между графическим, текстовым представлением SQL-оператора и результатами запроса, конструктор запросов утилиты SSMS способен одновременно отображать несколько областей, выбранных из контекстного меню (рис.8):

- область схемы (Pane | Diagram). В запросе может участвовать множество таблиц и представлений. Для их соединения связями с целью формирования предложения FROM оператора SELECT можно использовать это графическое представление;
- 

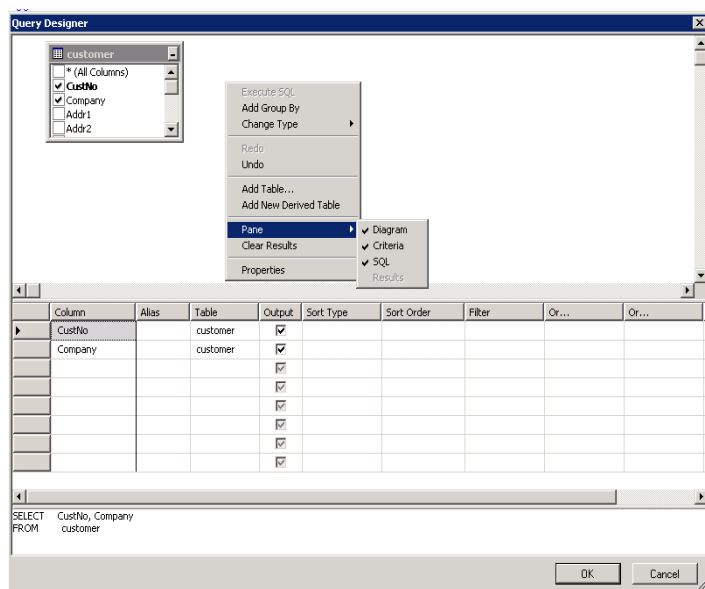


Рис.8. Конструктор запросов и его контекстное меню

область условия (Pane | Criteria). В этой области перечисляются отображаемые, фильтруемые, группируемые и сортируемые столбцы, а также условия фильтрации, группировки и сортировки;

- область SQL-кода (Pane | SQL). В этой области можно в текстовом виде ввести и отредактировать SQL-оператор;
- область результатов (Pane | Results). В этой области отображаются результаты выполнения запроса. Если результаты запроса остаются нетронутыми долгое время, утилита запрашивает у пользователя разрешение закрыть подключение к серверу.

Следует отметить, что, когда конструктор запросов открывается из редактора запросов, окно первого является модальным, а область результатов отключена.

Кроме оператора SELECT в конструкторе запросов можно формировать и другие операторы языка манипулирования данными (DML): INSERT, UPDATE и DELETE. Необходимый тип запроса указывается с помощью контекстного меню (рис.9).

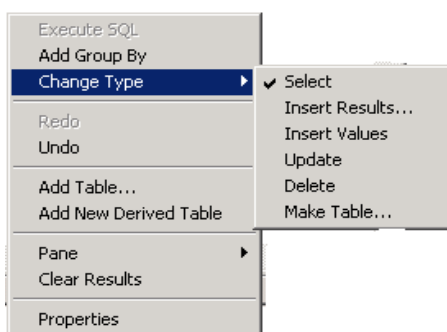


Рис.9. Команды контекстного меню, указывающие тип запроса

В конструкторе запросов область условия разделена на графы и строки. Каждая строка соответствует одному условию, которое задает критерий выбора данных из таблиц. Условия, записанные в нескольких строках, действуют одновременно, т.е. неявно связаны булевой операцией И (And).

Для оператора *SELECT* графы в области условия имеют следующее назначение (см. рис.8):

- Column - имя столбца таблицы, выбираемое из выпадающего списка, или выражение, набранное с помощью клавиатуры;
- Alias - псевдоним столбца, т.е. альтернативное имя, которое столбец будет иметь в таблице результатов запроса;
- Table - выбираемое из выпадающего списка имя таблицы базы данных, в которой содержится столбец, указанный в графе Column;
- Output - индикатор, обозначающий присутствие (галочка) или отсутствие (пусто) столбца в таблице результатов запроса;
- Sort Type - указатель типа сортировки строк таблицы результатов запроса по значениям столбца. Тип выбирается из выпадающего списка: Ascending - по возрастанию, Descending - по убыванию, Unsorted - без сортировки;
- Sort Order - порядковый номер столбца сортировки. Строки таблицы результатов запроса сортируются сначала по значениям столбца с номером 1, затем по значениям столбца с номером 2 и т.д. Номер может выбираться из выпадающего списка, в котором помимо чисел (1, 2, ...) предусмотрен элемент Unsorted, означающий, что по значениям столбца сортировка не происходит. При изменении порядкового номера у одного из столбцов сортировки автоматически осуществляется перенумерация других столбцов сортировки;
- Filter - условие, проверяемое для значений столбца или выражения, указанного в графе Column. Значения, для которых условие окажется истинным (True), указывают на необходимость включить данные из строк таблицы базы, содержащих эти значения, в таблицу результатов запроса;
- Or - дополнительное проверяемое условие, которое связывается булевой операцией ИЛИ с другими условиями, заданными в этой же строке.

Для автоматического заполнения ячеек строки, соответствующих графам Column, Table, Output, нужно в графическом представлении таблицы, находящемся в области схемы, щелчком мыши отметить требуемые столбцы таблицы.

Если в запросе используются данные из нескольких таблиц (такой запрос называется многотабличным), то эти таблицы необходимо соединить между собой с учетом имеющихся в них первичных и внешних ключей. В конструкторе запросов связь между таблицами задается в области схемы с помощью мыши: разместив курсор мыши на столбце одной из таблиц, нужно при нажатой левой кнопке мыши переместить курсор на соответствующий столбец другой таблицы и затем отпустить кнопку мыши (рис.10). Если связь между таблицами была задана при создании базы данных, то в области схемы эта связь отображается автоматически при добавлении таблиц в запрос.

Формирование запроса с группировкой, обеспечивающего образование групп строк и подведение итогов по каждой группе с использованием специальных агрегатных функций, задается командой контекстного меню Add Group By (см. рис.9), которая добавляет в область условия графу Group By (рис.11). В группу включаются строки таблицы, для которых значения в столбцах, называемых столбцами группировки, одинаковы. Чтобы столбец, указанный в графе Column, стал столбцом группировки, для него в графе Group By из выпадающего списка нужно выбрать элемент Group By. Агрегатная функция, выбираемая из выпадающего списка в графе Group By, подводит итог по каждой образовавшейся группе для указанного в графе Column столбца, например вычисляет сумму (Sum) значений, среднее (Avg) значение в указанном столбце или подсчитывает количество (Count) значений в столбце.

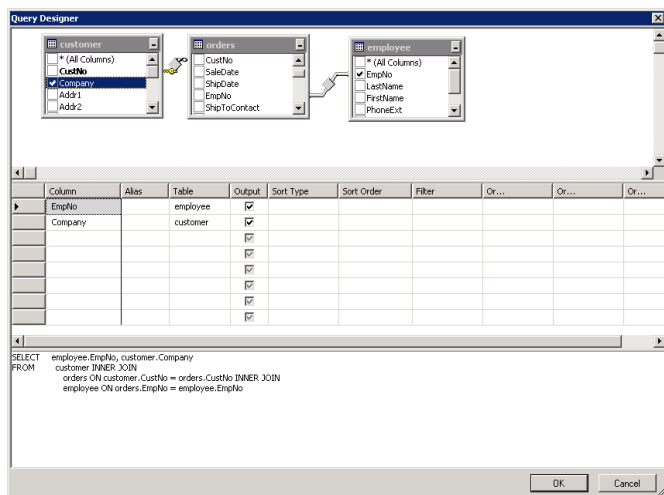


Рис.10. Связь таблиц в многотабличном запросе

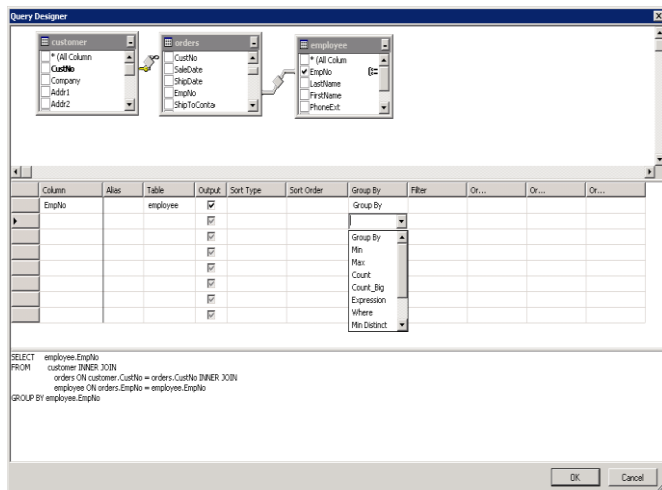


Рис.11. Средства формирования запроса с группировкой

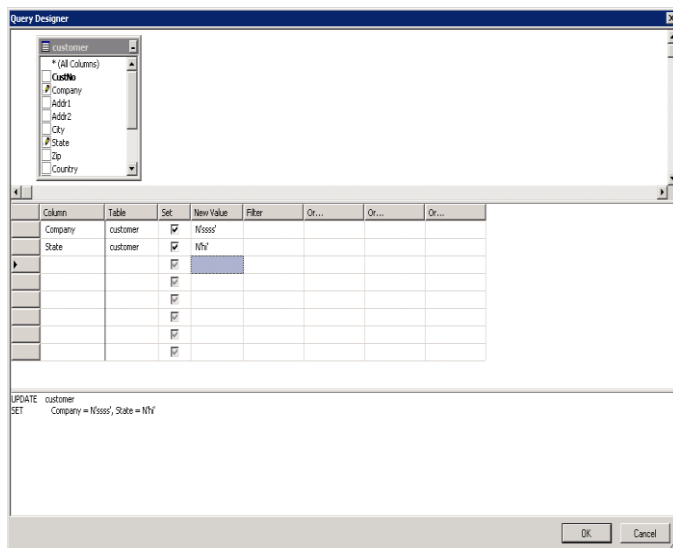


Рис.12. Средства формирования запроса обновления данных

Для обновления данных следует указать тип запроса Update (см. рис.9), которому соответствует область условия, показанная на рис.12. В графе New Value для столбца задается новое значение, а в последующих графах Filter и Or записываются условия поиска строк, в которых будет установлено новое значение в столбце.

Для добавления данных следует указать тип запроса Insert (см. рис.9), которому соответствует область условия, показанная на рис.13. В этой области для выбранного столбца таблицы, находящейся в области схемы, задается значение, которое будет помещено в строку, добавляемую в таблицу.

Для удаления данных следует указать тип запроса Delete (см. рис.9), которому соответствует область условия, показанная на рис.14. В этой области для таблицы, находящейся в области схемы, задаются условия, которым должны удовлетворять значения столбцов в строках, удаляемых из таблицы.

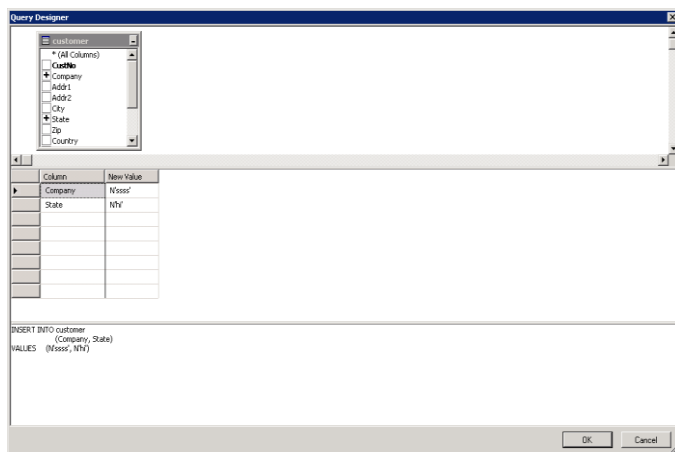


Рис.13. Средства формирования запроса добавления данных

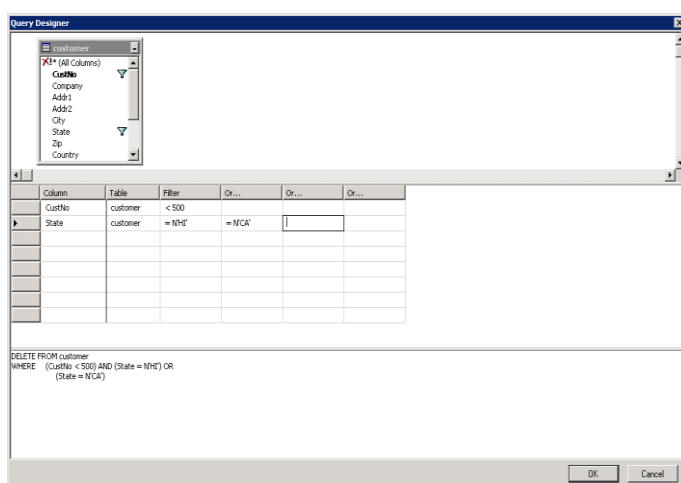


Рис.14. Средства формирования запроса удаления данных

## Лабораторное задание

1. При домашней подготовке изучить описание лабораторной работы и оформить пункты 1 и 2 отчета (см. требования к отчету).
2. При работе на компьютере ознакомиться с организацией пользовательского интерфейса утилиты SSMS.
3. С помощью конструктора запросов подготовить и выполнить запросы к таблицам демонстрационной базы данных DBdemosTest (табл.3-8), сохранив все сформированные SQL-запросы в одном файле сценария, т.е. в текстовом файле с расширением .sql. Для быстрого поиска запросов в файле сценария каждому запросу должен предшествовать комментарий, начинающийся двумя минусами и содержащий номер соответствующего пункта в разделе “Порядок выполнения работы”, а также словесную формулировку запроса, например:

-- 8 запрос на просмотр таблицы CUSTOMER

### Таблица 3

Структура таблицы customer (Покупатели)

№ п/п	Поле	Первичный ключ	Назначение
1	CustNo	*	Номер покупателя
2	Company		Название компании
...	...	...	...
5	City		Город
6	State		Штат
7	Zip		Почтовый индекс
...	...	...	...
10	FAX		Номер факса
11	TaxRate		Ставка налога (проценты)
12	Contact		Представитель компании
13	LastInvoiceDate		Дата последнего заказа

**Таблица 4**

#### Структура таблицы employee (Служащие)

№ п/п	Поле	Первичный ключ	Назначение
1	EmpNo	*	Номер служащего
2	LastName		Фамилия
3	FirstName		Имя
4	PhoneExt		Добавочный номер телефона
5	HireDate		Дата приема на работу
6	Salary		Жалование

**Таблица 5**

#### Структура таблицы orders (Заказы)

№ п/п	Поле	Первичный ключ	Назначение
1	OrderNo	*	Номер заказа
2	CustNo		Номер покупателя
3	SaleDate		Дата продажи
4	ShipDate		Дата поставки
5	EmpNo		Номер служащего
...	...	...	...
21	AmountPaid		Цена заказа



**Таблица 6****Структура таблицы items (Заказанные изделия)**

№ п/п	Поле	Первичный ключ	Назначение
1	OrderNo	*	Номер заказа
2	ItemsNo	*	Номер изделия в заказе
3	PartNo		Номер поставки партии изделий
4	Qty		Количество заказанных изделий
5	Discount		Скидка

**Структура таблицы parts (Поставки партий изделий)**

№ п/п	Поле	Первичный ключ	Назначение
1	PartNo	*	Номер поставки партии изделий
2	VendorNo		Номер поставщика
3	Description		Описание изделия
4	OnHand		Количество изделий в наличии
5	OnOrder		Количество заказанных изделий
6	Cost		Стоимость одного изделия
7	ListPrice		Цена одного изделия

**Таблица 8****Структура таблицы vendors (Поставщики)**

№ п/п	Поле	Первичный ключ	Назначение
1	VendorNo	*	Номер поставщика
2	VendorName		Название компании-поставщика
3	Address1		Адрес 1
4	Address2		Адрес 2
5	City		Город
6	State		Штат
7	Zip		Почтовый индекс
8	Country		Страна
9	Phone		Номер телефона
10	FAX		Номер факса

11	Preferred		Имеет предпочтение
----	-----------	--	--------------------

## Порядок выполнения работы

1. При домашней подготовке изучить описание лабораторной работы и письменно ответить на контрольные вопросы.

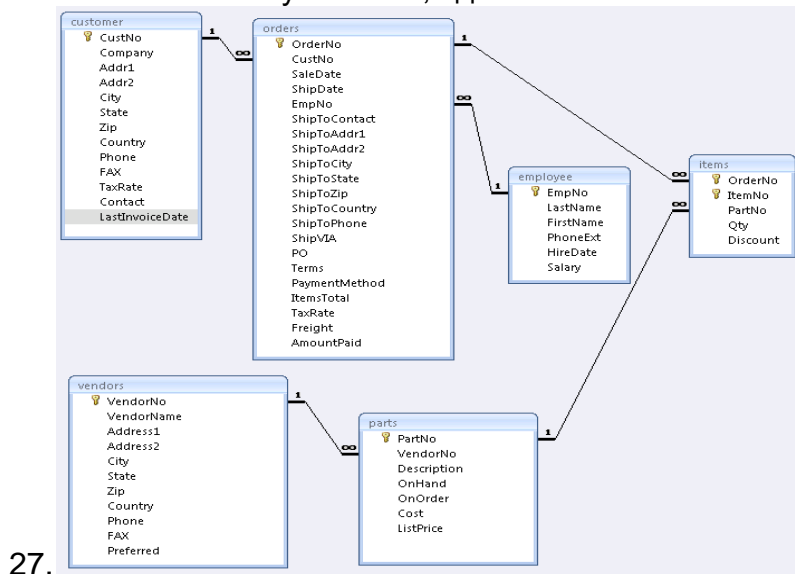


Рис.15. Окно соединения с сервером

2. Запустить утилиту SSMS через Терминал 4100 (skylab.sipc.miet.ru) командой Start | Microsoft SQL Server 2008 | SQL Server Management Studio. Для соединения с сервером задать тип сервера, имя сервера и режим аутентификации, показанные на рис.15, и нажать кнопку Connect.
3. При работе с утилитой SSMS с помощью команды главного меню View, кнопок стандартной панели инструментов или “горячих” клавиш открыть панели (окна) обозревателя объектов, сводки, зарегистрированных серверов, обозревателя шаблонов, свойств и редактора запросов. Ознакомиться с содержимым открытых окон.
4. Прикрепить окна обозревателя объектов и зарегистрированных серверов к левой границе главного окна утилиты, окна сводки и редактора запросов сделать частью документа, разделенного на вкладки, окна свойств и обозревателя шаблонов прикрепить к правой границе главного окна утилиты. Сделать окна свойств и зарегистрированных серверов автоматически скрываемыми.
5. В окне обозревателя объектов отобразить таблицы, входящие в базу данных DBdemosTest. В окнах сводки и обозревателя объектов для таблицы CUSTOMER показать перечень ее столбцов с указанием их типов.
6. Ознакомиться с диаграммой базы данных DBdemosTest, обратив внимание на связи таблиц.
7. В окнах сводки и обозревателя объектов отобразить таблицы, входящие в базу данных DBdemosTest и в имени которых встречается буквосочетание EM.
8. Активизировать редактор запросов и с помощью конструктора запросов сформировать запрос на просмотр таблицы CUSTOMER.
9. Ознакомиться с графическим представлением плана выполнения запроса и подробными сведениями о каждом этапе плана, подводя курсор к пиктограммам, обозначающим эти этапы.
10. В редакторе запросов набрать комментарий к сформированному запросу.
11. Командой File | Save As сохранить текст, содержащийся в окне редактора запросов, в файле U<№студбилета>.sql и открыть этот файл в Блокноте, чтобы копировать в него последующие прокомментированные запросы.

12. Выполнить несколько раз запрос на просмотр таблицы CUSTOMER, чтобы показать таблицу результатов запроса в табличном и текстовом формате.
13. С помощью конструктора запросов сформировать и выполнить запрос для получения таблицы результатов со столбцами Company и City, отсортированной в алфавитном порядке названий компаний. Скопировать сформированный запрос в файл U<Нестудбилета>.sql и сопроводить запрос комментарием. Не забывать копировать и комментировать и все следующие запросы.
14. С помощью конструктора запросов сформировать и выполнить запрос для получения таблицы результатов со столбцами Company и City, отсортированной в обратном алфавитном порядке названий компаний.
15. С помощью конструктора запросов сформировать и выполнить запрос для получения списка компаний-покупателей, расположенных в штате Гавайи (HI).
16. С помощью конструктора запросов сформировать и выполнить запрос для получения списка компаний-покупателей, расположенных в штате Гавайи (HI) или в Калифорнии (CA).
17. С помощью конструктора запросов сформировать и выполнить запрос для получения списка компаний-покупателей, сделавших заказ в январе 1995 года.
18. С помощью конструктора запросов сформировать и выполнить запрос для получения названий компаний-покупателей, начинающихся на букву "B" и оканчивающихся на "s". Для поиска названий по шаблону в качестве условия задается ключевое слово LIKE с последующим шаблоном - заключенной в апострофы строкой, в которой наряду с искомыми символам могут присутствовать подстановочные знаки "%" (обозначает любую последовательность символов) и "\_" (обозначает любой один символ).
19. С помощью конструктора запросов сформировать и выполнить запрос для получения списка компаний-покупателей с 5-значным почтовым индексом от 90000 до 99990.
20. С помощью конструктора запросов сформировать и выполнить запрос, который для всех компаний-покупателей вычисляет ставку налога TaxRate, увеличенную на 2%, без внесения изменений в таблицу.
21. В обозревателе объектов из контекстного меню таблицы CUSTOMER выбрать команду Select ... для вывода содержимого этой таблицы.
22. С помощью конструктора запросов сформировать и выполнить запрос с группировкой, который подсчитывает количество компаний-покупателей из каждого штата.
23. С помощью конструктора запросов сформировать и выполнить запрос, который для всех компаний-покупателей вычисляет ставку налога TaxRate, увеличенную на 1%, и вносит это изменение в таблицу.
24. В обозревателе объектов из контекстного меню таблицы CUSTOMER выбрать команду Select ... для вывода содержимого этой таблицы, чтобы убедиться в изменении ставки налога. Если отображается неизменная ставка налога, то следует нажать кнопку Refresh и повторно вывести содержимое таблицы.
25. С помощью конструктора запросов сформировать и выполнить запрос, который добавляет в таблицу CUSTOMER строку, указав в качестве значения поля CustNo целое число, составленное из номера студенческой группы и номера компьютера, и функцию Now( ) в качестве значения поля LastInvoiceDate.

26. С помощью конструктора запросов сформировать и выполнить запрос, который удаляет из таблицы CUSTOMER строки, соответствующие компаниям-покупателям, сделавшим заказы после 1 января 2011 года.



27.

Рис.16. Связь таблиц в демонстрационной базе данных

- С помощью конструктора запросов сформировать и выполнить многотабличный запрос (рис.16) для получения сведений о компаниях-покупателях, у которых цена заказа (столбец AmountPaid) меньше 1000 долларов.
28. С помощью конструктора запросов сформировать и выполнить многотабличный запрос для вывода названий компаний-покупателей, фамилий служащих, оформивших заказ, и имен поставщиков заказанных изделий. Ознакомиться с планом выполнения запроса.
29. С помощью конструктора запросов сформировать и выполнить многотабличный запрос для вывода названий компаний-покупателей и описаний изделий, заказанных в количестве, меньшем 10.
30. С помощью конструктора запросов сформировать и выполнить многотабличный запрос для вывода описаний изделий, их цены и названий компаний-покупателей, заказы которых не могут быть выполнены ( $OnHand < OnOrder$ ). Отсортировать строки таблицы результатов в порядке возрастания количества заказанных изделий и убыванию количества имеющихся изделий.
31. Закрыть файл сценария, ранее открытый в Блокноте и содержащий тексты SQL-запросов.
32. Открыть файл сценария в окне редактора запросов командой главного меню File | Open | File ... и показать преподавателю результаты выполнения запросов и вид запросов в конструкторе запросов.
33. Оформить отчет и защитить работу, ответив на заданные преподавателем вопросы.

## Требования к отчету

Отчет должен содержать:

1. название и цель работы;
2. ответы на контрольные вопросы;
3. файл сценария с текстом SQL-запросов и комментариями.

## Контрольные вопросы

1. Как запустить утилиту SSMS?
2. Какие режимы для управления размещением окон предусмотрены в утилите SSMS?
3. Как в утилите SSMS синхронно отобразить изменения, выполненные сервером в базе данных?
4. Как задать фильтрацию фрагмента дерева в окне обозревателя объектов? Для каких объектов возможна фильтрация?
5. Как активизировать редактор запросов?
6. Что называется пакетом в языке T-SQL и как отмечается его конец?
7. Какими способами можно передать серверу запрос или пакет для выполнения?
8. Как задать выполнение отдельного фрагмента пакета?
9. Как проверить синтаксическую правильность запроса?
10. Как указать базу данных, для которой будет выполнен пакет?
11. Для чего служит план выполнения запроса и как увидеть графическое представление плана?
12. Как открыть окно конструктора запросов и какие области имеются в нем?
13. Какие операторы языка T-SQL можно формировать в конструкторе запросов и как указать их тип?