Отчёт по практической работе № 8-9 студента Смирнова Константина группы ПКС-303

Тема: Создание сложных SQL запросов.

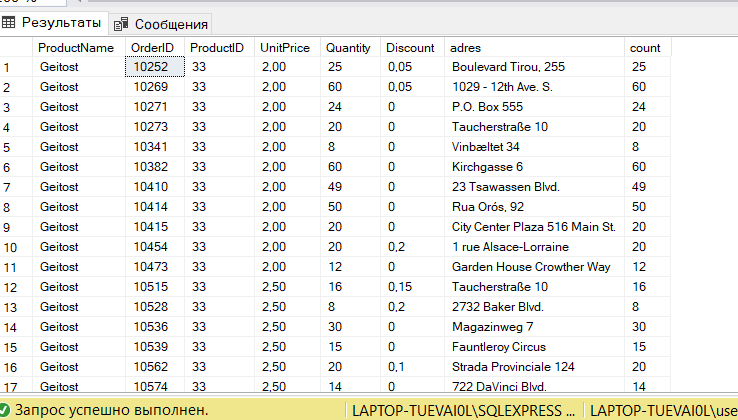
Цель работы: Научиться писать запросы к БД на языке SQL.

Выполнение работы

**Вариант 2**

Скачайте учебную базу данных (чётные номера по журналу - Northwind) или присоедините её. Создать SQL запросы, в которых:

1. Во введённом Вами заказе измените адрес и количество заказанного товара. (Такой пункт был в пр 5\_6. Сформировать заказ из ваших любимых товаров и вывести информацию о нём. Если Ваш заказ утерян, сначала повторите это задание).



UPDATE Orders SET ShipAddress='Les'

WHERE OrderID='10252'

UPDATE [Order Details] SET Quantity='30'

WHERE OrderID='10252'

SELECT DISTINCT ProductName, [Order Details].\*,ShipAddress as adres, Quantity as count

FROM dbo.Products INNER JOIN [Order Details] ON [Order Details].ProductID=Products.ProductID

INNER JOIN Orders ON Orders.OrderID = [Order Details].OrderID

WHERE ProductName LIKE 'Geitost'

Изображение выглядит как текст

Автоматически созданное описание

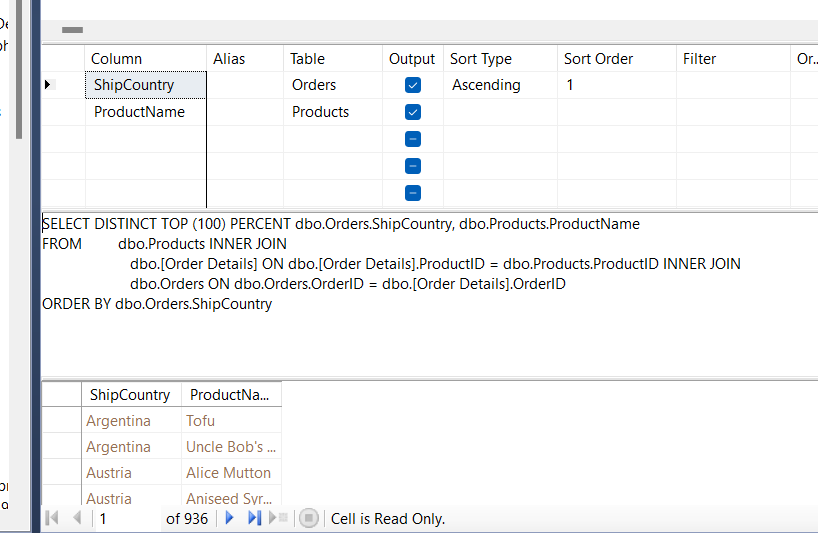
1. Вывести названия стран, которые являются заказчиками продуктов, список упорядочить по алфавиту.

SELECT DISTINCT ShipCountry,ProductName

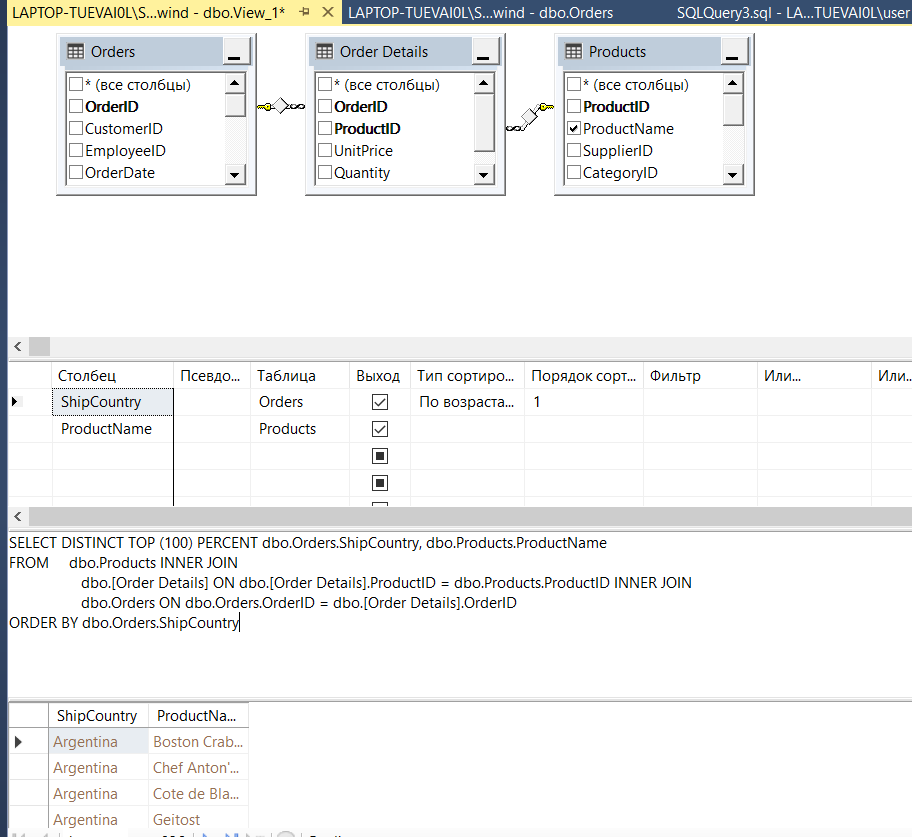
FROM dbo.Products INNER JOIN [Order Details] ON [Order Details].ProductID=Products.ProductID

INNER JOIN Orders ON Orders.OrderID = [Order Details].OrderID

ORDER BY ShipCountry

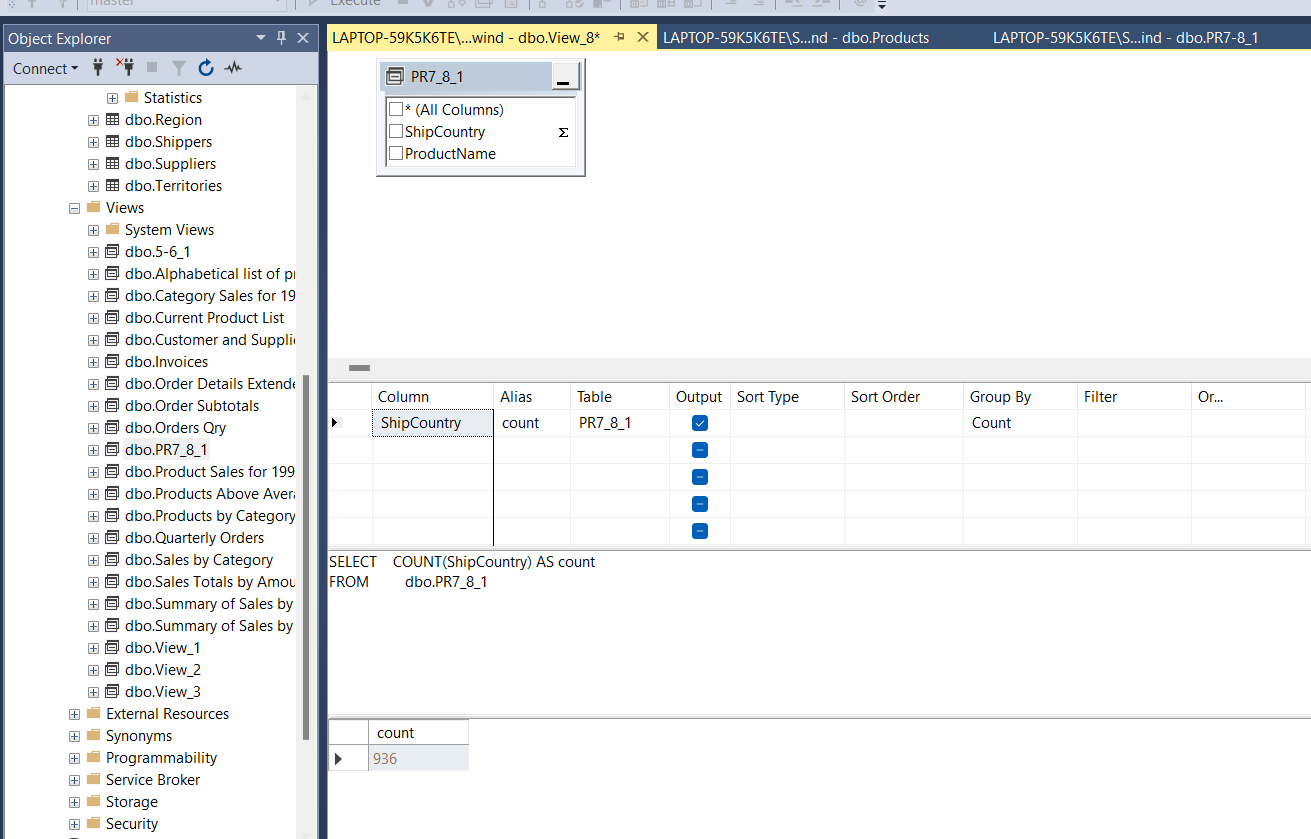


1. \*Вывести количество стран, которые являются заказчиками продуктов (использовать предыдущий запрос, который сохранить как представление).



SELECT count(ShipCountry) as count

FROM dbo.View\_1



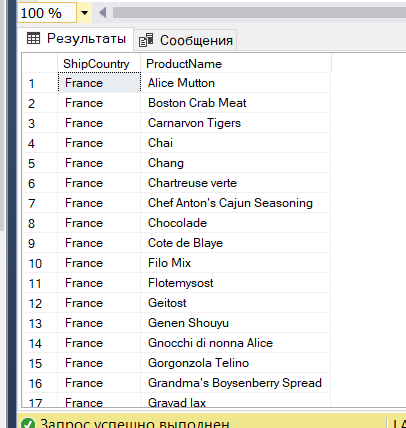
1. Выбрать список продуктов, заказанных страной, выбранной из предыдущего запроса по Вашему усмотрению.

SELECT DISTINCT dbo.Orders.ShipCountry, dbo.Products.ProductName

FROM dbo.Products INNER JOIN

dbo.[Order Details] ON dbo.[Order Details].ProductID = dbo.Products.ProductID INNER JOIN

dbo.Orders ON dbo.Orders.OrderID = dbo.[Order Details].OrderID

WHERE (dbo.Orders.ShipCountry = 'Austria')

1. Вывести **количество** продуктов, заказанных каждой страной, список упорядочить по убыванию количества продуктов.

SELECT DISTINCT ShipCountry, count(ProductName) as count

FROM dbo.Products INNER JOIN [Order Details] ON [Order Details].ProductID=Products.ProductID

INNER JOIN Orders ON Orders.OrderID = [Order Details].OrderID

GROUP BY ShipCountry

ORDER BY count(ProductName) DESC

Изображение выглядит как текст

Автоматически созданное описание

1. Вывести название страны, заказавшей больше (меньше) всего продуктов.

SELECT DISTINCT top 1 ShipCountry,count(ProductName) as count

FROM dbo.Products INNER JOIN [Order Details] ON [Order Details].ProductID=Products.ProductID

INNER JOIN Orders ON Orders.OrderID = [Order Details].OrderID

GROUP BY ShipCountry

ORDER BY count(ProductName) DESC

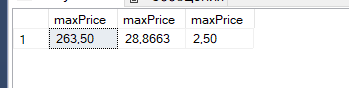
Изображение выглядит как текст

Автоматически созданное описание

1. Вывести минимальную, максимальную и среднюю цену товаров.

SELECT max(UnitPrice) as maxPrice,avg(UnitPrice) as maxPrice,min(UnitPrice) as maxPrice

FROM dbo.Products



1. Вывести **название товаров**, которые имеют цену ниже средней.

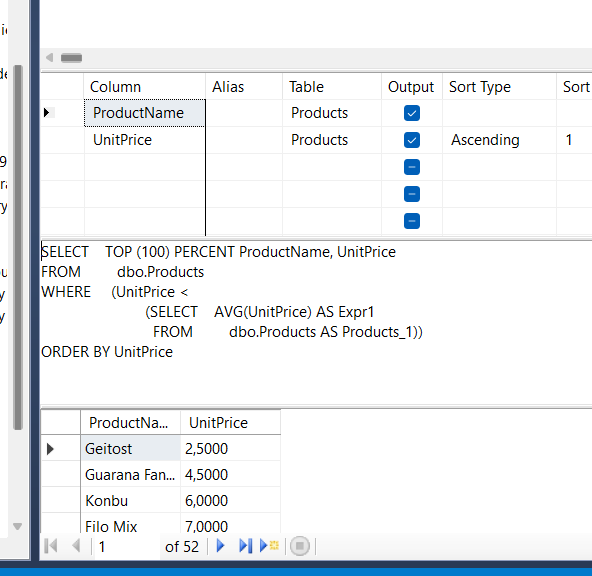
SELECT ProductName,UnitPrice

FROM dbo.Products

WHERE UnitPrice < (SELECT avg(UnitPrice)

FROM dbo.Products)

ORDER BY UnitPrice



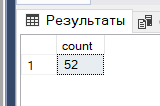
1. \*Вывести **количество** товаров, которые имеют цену ниже средней.

SELECT count(ProductName) as count

FROM dbo.Products

WHERE UnitPrice < (SELECT avg(UnitPrice)

FROM dbo.Products)



1. Вывести номер заказа, название товара, цену, количество, скидку, стоимость товара в заказе с учетом скидки и без нее (Использовать таблицы Детали\_заказа и Продукты). Выбрать любой заказ, в котором заказано более 2-х товаров.

SELECT [Order Details].OrderID,ProductName,[Order Details].UnitPrice,Quantity,Discount,

([Order Details].UnitPrice\*Quantity)/100\*discount as discount\_amount,

[Order Details].UnitPrice\*Quantity as price\_without\_discount, ([Order Details].UnitPrice\*Quantity)-(([Order Details].UnitPrice\*Quantity)/100\*discount) as discounted\_price

FROM dbo.Products INNER JOIN [Order Details] ON [Order Details].ProductID=Products.ProductID

WHERE Quantity>2

Изображение выглядит как стол

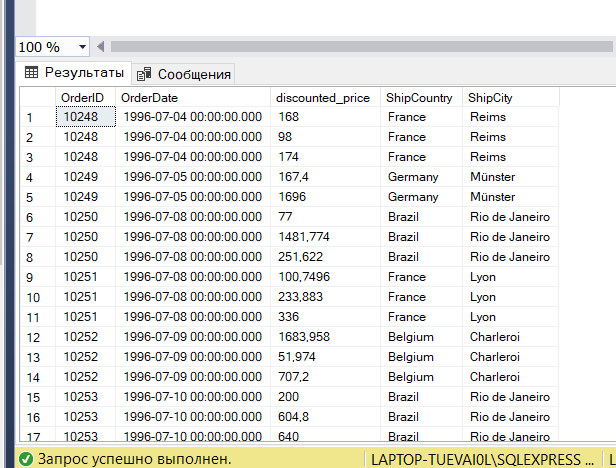
Автоматически созданное описание

1. Вывести номер заказа, дату заказа и сумму заказа с учетом скидки, страну и город Заказчика.

SELECT [Order Details].OrderID,OrderDate, ([Order Details].UnitPrice\*Quantity)-(([Order Details].UnitPrice\*Quantity)/100\*discount) as discounted\_price,ShipCountry,ShipCity

FROM dbo.Products INNER JOIN [Order Details] ON [Order Details].ProductID=Products.ProductID

INNER JOIN Orders ON Orders.OrderID=[Order Details].OrderID

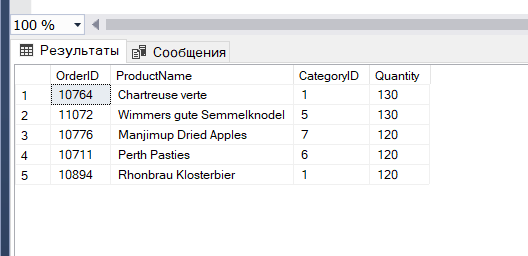


1. \* Вывести категории продуктов, которые заказывают чаще всего.

SELECT top 5 OrderID,ProductName,CategoryID,Quantity

FROM dbo.Products INNER JOIN [Order Details] ON [Order Details].ProductID=Products.ProductID

ORDER BY Quantity DESC



1. \* Вывести **количество** продуктов, которые заказала страна, выбранная по Вашему усмотрению.

SELECT DISTINCT dbo.Orders.ShipCountry, COUNT(dbo.Products.ProductName) AS count

FROM dbo.Products INNER JOIN

dbo.[Order Details] ON dbo.[Order Details].ProductID = dbo.Products.ProductID INNER JOIN

dbo.Orders ON dbo.Orders.OrderID = dbo.[Order Details].OrderID

WHERE (dbo.Orders.ShipCountry = 'Austria')

GROUP BY dbo.Orders.ShipCountry

Изображение выглядит как текст

Автоматически созданное описание

1. \*Вывести **названия** продуктов, которые **не** заказала страна, выбранная по Вашему усмотрению.

SELECT DISTINCT ShipCountry,ProductName

FROM dbo.Products INNER JOIN [Order Details] ON [Order Details].ProductID=Products.ProductID

INNER JOIN Orders ON Orders.OrderID = [Order Details].OrderID

WHERE ShipCountry='Poland'

SELECT ProductName

FROM dbo.Products P

WHERE NOT EXISTS

(

SELECT 1

FROM dbo.Products Left outer JOIN

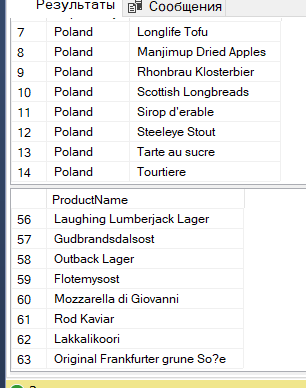
dbo.[Order Details] ON dbo.[Order Details].ProductID = dbo.Products.ProductID Left outer JOIN

dbo.Orders ON dbo.Orders.OrderID = dbo.[Order Details].OrderID

/\*FROM View\_2 I\*/

WHERE ShipCountry='Poland' and [Order Details].ProductID = P.ProductID

)



1. \* Вывести **количество** продуктов, которые **не** заказала страна, выбранная по Вашему усмотрению.

SELECT DISTINCT ShipCountry,ProductName

FROM dbo.Products INNER JOIN [Order Details] ON [Order Details].ProductID=Products.ProductID

INNER JOIN Orders ON Orders.OrderID = [Order Details].OrderID

WHERE ShipCountry='Austria'

SELECT ProductName

FROM dbo.Products P

WHERE NOT EXISTS

(

SELECT 1

FROM dbo.Products Left outer JOIN

dbo.[Order Details] ON dbo.[Order Details].ProductID = dbo.Products.ProductID Left outer JOIN

dbo.Orders ON dbo.Orders.OrderID = dbo.[Order Details].OrderID

/\*FROM View\_2 I\*/

WHERE ShipCountry='Austria' and [Order Details].ProductID = P.ProductID

)

Details].ProductID = P.ProductID

)



1. \*\*Вывести сумму заказов каждой страны по годам в виде: название страны год сумма\_заказов\_за\_год.

SELECT DISTINCT ShipCountry, YEAR(OrderDate) as year,sum([Order Details].OrderID) as sum

FROM dbo.Products INNER JOIN [Order Details] ON [Order Details].ProductID=Products.ProductID

INNER JOIN Orders ON Orders.OrderID = [Order Details].OrderID

GROUP BY ShipCountry,YEAR(OrderDate)

ORDER BY ShipCountry



1. \*\* Вывести год с максимальной суммой заказов по всем странам.

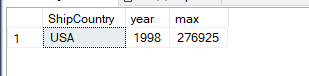
SELECT top 1 ShipCountry,YEAR(OrderDate) as year,sum(Orders.OrderID) as max

FROM dbo.Products INNER JOIN [Order Details] ON [Order Details].ProductID=Products.ProductID

INNER JOIN Orders ON Orders.OrderID = [Order Details].OrderID

GROUP BY ShipCountry,YEAR(OrderDate),Orders.OrderID

ORDER BY max DESC



1. Удалите Ваш заказ из базы данных.

DELETE FROM products

WHERE productID='78'

Изображение выглядит как текст

Автоматически созданное описание

Ответы на контрольные вопросы

1. Как устранить дублирование записей?

С помощью целостности данных и нормальный форм

1. Назовите минимальный синтаксис команд: SELECT, DELETE, UPDATE, INSERT INTO.

— SELECT [ALL | DISTINCT] <список\_выбора>

— DELETE FROM таблица

— UPDATE таблица SET { поле=значение }

— INSERT INTO таблица [ ( поле [,...] ) ]

1. Назовите виды встроенных функций MS SQL Server, приведите примеры их использования.

Агрегатные функции, функции конфигурации, функции работы с курсорами, функции даты и времени, математические функции, функции метаданных, прочие функции, функции идентификаторов иерархии, функции наборов строк, функции безопасности, строковые функции, системные статические функции, функции работы с текстами и изображениями.

Строковая функция len - Возвращает количество символов указанного строкового выражения, исключая конечные пробелы.

Строковая функция str - Возвращает символьные данные, преобразованные из числовых данных. Символьные данные выровнены по правому краю с заданной длиной и десятичной точностью.

Функция даты и времени day - Эта функция возвращает целое число, представляющее дату (день месяца) указанного значения типа date.

1. Каким образом сделать сортировку записей в таблице?

Для сортировки данных, получаемых в результате запроса в языке SQL используется ключевое слово ORDER BY. База данных (далее – БД) сортирует данные по указанному параметру по убыванию, либо по возрастанию, в зависимости от условия указанного в запросе (ASC / DESC – соответственно).

1. Назовите виды встроенных функций MS SQL Server, приведите примеры их использования.

внутреннее соединение,

левое внешнее соединение.

Правое внешнее соединение

Полное внешнее соединение

Перекрестное соединение

1. Чем отличается операция соединения от операции объединения?

В результате операции UNION сцепляются результирующие наборы двух запросов. При этом операция UNION не создает отдельные строки для столбцов, полученных из двух таблиц. Операция JOIN сравнивает столбцы из двух таблиц и создает результирующие строки, которые состоят из столбцов из двух таблиц.

1. Соединили 2 таблицы, в которых есть записи, в результирующей таблице нет записей, возможно ли такое, если, да, то когда?

Да, в случае, если в одной таблице есть та информация, чего нет в другой, и при этом, если в запросе сделано inner join, left join

1. К полям какого типа применяются функции Count, Max, Min,Avg, Sum?

sum и avg - только к числовым, а cout, max и min могут использоваться числовые или символьные поля. При использовании с символьными полями MAX, MIN сравнивают значения в алфавитном порядке.

1. Сформулируйте правило ссылочной целостности.

Значение внешнего ключа должно соответствовать значению первичного ключа или должно быть пустым.

1. Чем отличается HAVING от GROUP BY?

HAVING - применяется для фильтрации функций и столбцов сгруппированных при помощи GROUP BY указанных в SELECT. Другими словами применяется для агрегатных функций(COUNT(), MAX() ...) и столбцов указанных в выражении SELECT и обработанных GROUP BY.

1. Чем отличается SQL от языков программирования?

SQL – это язык программирования декларативного типа. В отличие от привычных нам процедурных языков, в которых есть условия, циклы и функции, в декларативных языках подобных алгоритмических конструкций почти нет. Декларативные выражения представляют собой скорее запросы, описание того, что хочет получить человек.

1. Назовите команды DDL, DML, DCL, TCL.

Data Definition Language (DDL) – это группа операторов определения данных.

* CREATE – используется для создания объектов базы данных;
* ALTER – используется для изменения объектов базы данных;
* DROP – используется для удаления объектов базы данных.

Data Manipulation Language (DML) – это группа операторов для манипуляции данными.

* SELECT – осуществляет выборку данных;
* INSERT – добавляет новые данные;
* UPDATE – изменяет существующие данные;
* DELETE – удаляет данные.

Data Control Language (DCL) – группа операторов определения доступа к данным.

* GRANT – предоставляет пользователю или группе разрешения на определённые операции с объектом;
* REVOKE – отзывает выданные разрешения;
* DENY– задаёт запрет, имеющий приоритет над разрешением.

Transaction Control Language (TCL) – группа операторов для управления транзакциями.

* BEGIN TRANSACTION – служит для определения начала транзакции;
* COMMIT TRANSACTION – применяет транзакцию;
* ROLLBACK TRANSACTION – откатывает все изменения, сделанные в контексте текущей транзакции;
* SAVE TRANSACTION – устанавливает промежуточную точку сохранения внутри транзакции.

1. Что должен уметь делать системный аналитик?

Системный аналитик разрабатывает требования к программному обеспечению. Заказчик или владелец продукта определяет, что должна делать программа. После этого системный аналитик общается с заказчиком или владельцем продукта, добывает информацию из разных источников и изучает рынок.

1. Можно ли одной командой UPDATE обновить несколько полей в группе записей?

Если инструкция UPDATE могла обновить несколько строк при обновлении как ключа кластеризации, так и одного или нескольких столбцов типа text, ntext или image, то частичное обновление этих столбцов выполняется как полная замена значений. (хз, то или нет)

1. Что такое каскадное обновление связанных записей?

Если при включении обеспечения целостности данных был включен параметр "Каскадное обновление связанных полей", то при последующем обновлении первичного ключа автоматически будут обновлены все связанные с ним поля.

**Выводы о полученных практических навыках**

Я получил опыт работы с запросами в MS SQL SERVER.

**Дата \_\_\_\_\_\_\_\_\_\_\_\_\_ оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ подпись**

**замечание преподавателя**