

# MS SQL Server разработчик

Guru SQL: подробно об оптимизации, планах, транзакциях и о том, как писать эффективные запросы на данных большого объёма

Длительность курса: 152 академических часа

## 1 Использование БД

### 1 Базы данных и какие они бывают

узнать про разные виды систем управления баз данных и самих БД  
уметь выбрать где и какой вид лучше использовать  
узнать про новые возможностях SQL server  
развернуть бэкап учебной базы

Домашние задания

1 Разворачиваем Backup и смотрим какой проект какую СУБД использует

1. Установите программы SQL Server 2016\2017
2. Разверните у себя бэкап базы WideWorldImporters (скриншот с восстановленной БД) WideWorldImporters-Full.bak

3. Поставьте SQL Sentry Plan Explorer

4. Сделайте проект для курса на github, пришлите ссылку на него (тут будем выкладывать скрипты по домашкам)

5. Придумайте и сделайте описание проекта, который будете делать в рамках всего курса. В качестве проекта вы можете взять любую идею, которая вам близка. Это может быть какая-то часть вашего рабочего проекта, которую вы хотите переосмыслить.

Если есть идея, но не понятно как ее уложить в рамки учебного проекта, напишите преподавателю и мы поможем.

Проект мы будем делать весь курс и защищать его в самом конце, он будет заключаться в созданной БД со схемой, описанием проекта, и необходимыми процедурами\функциями или SQL кодом для демонстрации основного функционала системы.

Пока достаточно создать в github папку с проектом, создать там описание проекта - о чем он, какие функции будут реализованы, основные сущности, которые затем будут созданы (просто описание текстом).

Опционально

6. Найдите, какую СУБД использует любимый вами проект

Выберите проект, которым вы часто пользуетесь и найдите информацию с помощью какой базы данных он реализован.

Классификация: иерархическая, сетевая, реляционная, объектно-ориентированная, NoSql (если знаете колоночная, key-value, документарная, in-memory, графовая)

Структура ответа:

Название проекта: Проект

Используемые базы данных: БД (реляционная), БД (NoSQL документарная)

Количество серверов БД: 10

Ссылка на источник:

## 2 Оператор SELECT и простые фильтры

научиться делать выборки используя оператор SELECT  
использовать простые фильтры в WHERE и объединение условий  
попробовать разные виды JOIN  
разобраться с тем как читать планы запросов

Домашние задания

### 1 Запросы SELECT

Напишите выборки для того, чтобы получить:

1. Все товары, в которых в название есть пометка urgent или название начинается с Animal
  2. Поставщиков, у которых не было сделано ни одного заказа (потом покажем как это делать через подзапрос, сейчас сделайте через JOIN)
  3. Продажи с названием месяца, в котором была продажа, номером квартала, к которому относится продажа, включите также к какой трети года относится дата - каждая треть по 4 месяца, дата забора заказа должна быть задана, с ценой товара более 100\$ либо количество единиц товара более 20. Добавьте вариант этого запроса с постраничной выборкой пропустив первую 1000 и отобразив следующие 100 записей. Сортировка должна быть по номеру квартала, трети года, дате продажи.
  4. Заказы поставщикам, которые были исполнены за 2014й год с доставкой Road Freight или Post, добавьте название поставщика, имя контактного лица принимавшего заказ
  5. 10 последних по дате продаж с именем клиента и именем сотрудника, который оформил заказ.
  6. Все ид и имена клиентов и их контактные телефоны, которые покупали товар Chocolate frogs 250g
-

писать запросы с использованием подзапросов  
рассказать, что будет в результате выполнения  
запроса, без его запуска

## Домашние задания

### 1 Подзапросы и CTE

Для всех заданий где возможно, сделайте 2  
варианта запросов:

- 1) через вложенный запрос
- 2) через WITH (для производных таблиц)

Напишите запросы:

1. Выберите сотрудников, которые являются  
продажниками, и еще не сделали ни одной  
продажи.
2. Выберите товары с минимальной ценой  
(подзапросом), 2 варианта подзапроса.
3. Выберите информацию по клиентам, которые  
перевели компании 5 максимальных платежей из  
[Sales].[CustomerTransactions] представьте 3  
способа (в том числе с CTE)
4. Выберите города (ид и название), в которые  
были доставлены товары, входящие в тройку  
самых дорогих товаров, а также Имя сотрудника,  
который осуществлял упаковку заказов
5. Объясните, что делает и оптимизируйте  
запрос:

```
SELECT
Invoices.InvoiceID,
Invoices.InvoiceDate,
(SELECT People.FullName
FROM Application.People
WHERE People.PersonID =
Invoices.SalespersonPersonID
) AS SalesPersonName,
SalesTotals.TotalSumm AS TotalSummByInvoice,
(SELECT
SUM(OrderLines.PickedQuantity*OrderLines.UnitPri
ce)
FROM Sales.OrderLines
```

```

WHERE OrderLines.OrderId = (SELECT
Orders.OrderId
FROM Sales.Orders
WHERE Orders.PickingCompletedWhen IS NOT
NULL
AND Orders.OrderId = Invoices.OrderId)
) AS TotalSummForPickedItems
FROM Sales.Invoices
JOIN
(SELECT InvoicId, SUM(Quantity*UnitPrice) AS
TotalSumm
FROM Sales.InvoiceLines
GROUP BY InvoicId
HAVING SUM(Quantity*UnitPrice) > 27000) AS
SalesTotals
ON Invoices.InvoiceID = SalesTotals.InvoicId
ORDER BY TotalSumm DESC

```

Приложите план запроса и его анализ, а также ход ваших рассуждений по поводу оптимизации. Можно двигаться как в сторону улучшения читабельности запроса (что уже было в материале лекций), так и в сторону упрощения плана\ускорения.

Опциональная часть:

В материалах к вебинару есть файл HT\_reviewBigCTE.sql - прочтите этот запрос и напишите что он должен вернуть и в чем его смысл, можно если есть идеи по улучшению тоже их включить.

---

**4 Введение в планы запросов**

читать и объяснять план запроса  
оценивать разные запросы и сравнивать их стоимость

---

**5 GROUP BY и HAVING. Временные таблицы и**

писать запрос с использованием GROUP BY и HAVING,  
писать запрос с Cube и Rollup  
перечислить отличие HAVING и WHERE

## Домашние задания

### 1 Группировки и агрегатные функции

1. Посчитать среднюю цену товара, общую сумму продажи по месяцам
2. Отобразить все месяцы, где общая сумма продаж превысила 10 000
3. Вывести сумму продаж, дату первой продажи и количество проданного по месяцам, по товарам, продажи которых менее 50 ед в месяц. Группировка должна быть по году и месяцу.
4. Написать рекурсивный CTE sql запрос и заполнить им временную таблицу и табличную переменную

Дано :

```
CREATE TABLE dbo.MyEmployees
(
EmployeeID smallint NOT NULL,
FirstName nvarchar(30) NOT NULL,
LastName nvarchar(40) NOT NULL,
Title nvarchar(50) NOT NULL,
DeptID smallint NOT NULL,
ManagerID int NULL,
CONSTRAINT PK_EmployeeID PRIMARY KEY
CLUSTERED (EmployeeID ASC)
);
INSERT INTO dbo.MyEmployees VALUES
(1, N'Ken', N'Sánchez', N'Chief Executive
Officer',16,NULL)
,(273, N'Brian', N'Welcker', N'Vice President of
Sales',3,1)
,(274, N'Stephen', N'Jiang', N'North American Sales
Manager',3,273)
,(275, N'Michael', N'Blythe', N'Sales
Representative',3,274)
,(276, N'Linda', N'Mitchell', N'Sales
Representative',3,274)
```

),(285, N'Syed', N'Abbas', N'Pacific Sales  
Manager',3,273)  
,(286, N'Lynn', N'Tsoflias', N'Sales  
Representative',3,285)  
,(16, N'David',N'Bradley', N'Marketing Manager', 4,  
273)  
,(23, N'Mary', N'Gibson', N'Marketing Specialist', 4,  
16);

Результат вывода рекурсивного CTE:

EmployeeID	Name	Title	EmployeeLevel
1	Ken Sánchez	Chief Executive Officer	1
273	Brian Welcker	Vice President of Sales	2
16	David Bradley	Marketing Manager	3
23	Mary Gibson	Marketing Specialist	4
274	Stephen Jiang	North American Sales Manager	3
276	Linda Mitchell	Sales Representative	4
275	Michael Blythe	Sales Representative	4
285	Syed Abbas	Pacific Sales Manager	3
286	Lynn Tsoflias	Sales Representative	4

Опционально:

Написать все эти же запросы, но, если за какой-то месяц не было продаж, то этот месяц тоже должен быть в результате и там должны быть нули.

---

## 6 SQL операторы изменения данных

уметь определить к чему относится оператор к DDL или DML  
использовать операторы INSERT, UPDATE, UPSERT, DELETE, MERGE, Bulk insert  
использовать утилиту bcp

Домашние задания

### 1 Insert, Update, Merge

1. Довставлять в базу 5 записей используя insert в таблицу Customers или Suppliers
2. удалите 1 запись из Customers, которая была вами добавлена
3. изменить одну запись, из добавленных через UPDATE
4. Написать MERGE, который вставит запись в клиенты, если ее там нет, и изменит если она уже есть
5. Напишите запрос, который выгрузит данные через bcp out и загрузить через bulk insert

## 7 Оконные функции в SQL

писать запросы с использованием оконных функций  
проходить тест по оконным функциям

Домашние задания

### 1 Оконные функции

1. Напишите запрос с временной таблицей и перепишите его с табличной переменной. Сравните планы.  
В качестве запроса с временной таблицей и табличной переменной можно взять свой запрос или следующий запрос:  
Сделать расчет суммы продаж нарастающим итогом по месяцам с 2015 года (в рамках одного месяца он будет одинаковый, нарастать будет в течение времени выборки)  
Выведите id продажи, название клиента, дату

продажи, сумму продажи, сумму нарастающим  
итогом

Пример

Дата продажи Нарастающий итог по месяцу

2015-01-29 4801725.31

2015-01-30 4801725.31

2015-01-31 4801725.31

2015-02-01 9626342.98

2015-02-02 9626342.98

2015-02-03 9626342.98

Продажи можно взять из таблицы Invoices.

Нарастающий итог должен быть без оконной  
функции.

2. Если вы брали предложенный выше запрос, то  
сделайте расчет суммы нарастающим итогом с  
помощью оконной функции.

Сравните 2 варианта запроса - через windows  
function и без них. Написать какой быстрее  
выполняется, сравнить по set statistics time on;

2. Вывести список 2х самых популярных  
продуктов (по кол-ву проданных) в каждом  
месяце за 2016й год (по 2 самых популярных  
продукта в каждом месяце)

3. Функции одним запросом

Посчитайте по таблице товаров, в вывод также  
должен попасть ид товара, название, брэнд и  
цена

пронумеруйте записи по названию товара, так  
чтобы при изменении буквы алфавита нумерация  
начиналась заново

посчитайте общее количество товаров и

выведете полем в этом же запросе

посчитайте общее количество товаров в

зависимости от первой буквы названия товара

отобразите следующий id товара исходя из того,

что порядок отображения товаров по имени

предыдущий id товара с тем же порядком

отображения (по имени)

названия товара 2 строки назад, в случае если

предыдущей строки нет нужно вывести "No items"

сформируйте 30 групп товаров по полю вес

товара на 1 шт

Для этой задачи НЕ нужно писать аналог без аналитических функций

4. По каждому сотруднику выведите последнего клиента, которому сотрудник что-то продал  
В результатах должны быть ид и фамилия сотрудника, ид и название клиента, дата продажи, сумму сделки

5. Выберите по каждому клиенту 2 самых дорогих товара, которые он покупал

В результатах должно быть ид клиента, его название, ид товара, цена, дата покупки

Опционально можно сделать вариант запросов для заданий 2,4,5 без использования windows function и сравнить скорость как в задании 1.

Bonus из предыдущей темы

Напишите запрос, который выбирает 10 клиентов, которые сделали больше 30 заказов и последний заказ был не позднее апреля 2016.

## 8 Операторы CROSS APPLY, PIVOT, CUBE

писать запросы с использованием CROSS APPLY, PIVOT и UNPIVOT, описывать значение операторов

Домашние задания

### 1 Pivot и Cross Apply

1. Требуется написать запрос, который в результате своего выполнения формирует таблицу следующего вида:

Название клиента

Месяц Год Количество покупок

Клиентов взять с ID 2-6, это все подразделение Tailspin Toys

имя клиента нужно поменять так чтобы осталось только уточнение

например исходное Tailspin Toys (Gasport, NY) - вы выводите в имени только Gasport, NY

дата должна иметь формат dd.mm.yyyy  
например 25.12.2019

Например, как должны выглядеть результаты:  
InvoiceMonth Peoples Valley, AZ Medicine Lodge,  
KS Gasport, NY Sylvanite, MT Jessie, ND  
01.01.2013 3 1 4 2 2  
01.02.2013 7 3 4 2 1

2. Для всех клиентов с именем, в котором есть  
Tailspin Toys  
вывести все адреса, которые есть в таблице, в  
одной колонке

Пример результатов

CustomerName AddressLine  
Tailspin Toys (Head Office) Shop 38  
Tailspin Toys (Head Office) 1877 Mittal Road  
Tailspin Toys (Head Office) PO Box 8975  
Tailspin Toys (Head Office) Ribeiroville  
.....

3. В таблице стран есть поля с кодом страны  
цифровым и буквенным  
сделайте выборку ИД страны, название, код -  
чтобы в поле был либо цифровой либо буквенный  
код

Пример выдачи

CountryId CountryName Code  
1 Afghanistan AFG  
1 Afghanistan 4  
3 Albania ALB  
3 Albania 8

4. Перепишите ДЗ из оконных функций через  
CROSS APPLY  
Выберите по каждому клиенту 2 самых дорогих  
товара, которые он покупал  
В результатах должно быть ид клиента, его  
название, ид товара, цена, дата покупки

5. Code review (опционально). Запрос приложен в материалы Hometask\_code\_review.sql.

Что делает запрос?

Чем можно заменить CROSS APPLY - можно ли использовать другую стратегию выборки\запроса?

---

## 9 Операторы DDL

создавать объекты БД с помощью своих SQL запросов

Домашние задания

### 1 Используем DDL

Начало проектной работы.

Создание таблиц и представлений для своего проекта.

Нужно используя операторы DDL создать:

1. 3-4 основные таблицы для своего проекта.
2. Первичные и внешние ключи для всех созданных таблиц
3. 1-2 индекса на таблицы

В качестве проекта вы можете взять любую идею, которая вам близка и сделать схему базы данных, а затем создать ее.

Это может быть какая-то часть вашего рабочего проекта, которую вы хотите переосмыслить.

Если есть идея, но не понятно как ее уложить в рамки учебного проекта, напишите преподавателю и мы поможем.

---

## 10 Итоговое занятие по SELECT

перечислять этапы выполнения запроса  
читать длинные запросы и рассказывать что должно получиться после выполнения, без выполнения запроса

---

- 11 **Динамический SQL** создавать процедуры с использованием динамического SQL, перечислять ошибки при использовании динамического SQL которые ведут к угрозам безопасности  
объяснить как можно сохранять план запроса по динамическому SQL
- 
- 12 **Выборки из xml и json полей** писать выборки из xml полей  
писать выборки из json полей
- Домашние задания
- 1 XML, JSON и динамический SQL
1. Загрузить данные из файла StockItems.xml в таблицу StockItems.  
Существующие записи в таблице обновить, отсутствующие добавить (искать по StockItemName).  
Файл StockItems.xml в личном кабинете.
  2. Выгрузить данные из таблицы StockItems в такой же xml-файл, как StockItems.xml
  3. В таблице StockItems в колонке CustomFields есть данные в json.  
Написать select для вывода:
    - StockItemID
    - StockItemName
    - CountryOfManufacture (из CustomFields)
    - Range (из CustomFields)
  4. Найти в StockItems строки, где есть тэг "Vintage"  
Запрос написать через функции работы с JSON.  
Тэги искать в поле CustomFields, а не в поле Tags.  
Для поиска использовать равенство, использовать LIKE запрещено.

Запрос должен быть примерно в таком виде:  
SELECT ... WHERE ... = 'Vintage'

## 5. Пишем динамический PIVOT.

По заданию из 8го занятия про CROSS APPLY и PIVOT

Требуется написать запрос, который в результате своего выполнения формирует таблицу следующего вида:

Название клиента

Месяц Год Количество покупок

Нужно написать запрос, который будет генерировать результаты для всех клиентов имя клиента указывать полностью из CustomerName  
дата должна иметь формат dd.mm.yyyy  
например 25.12.2019

## 13 Хранимые процедуры и функции

создавать хранимые процедуры и функции  
объяснять различия  
объяснять хранение плана запроса по процедурам, функциям и запросам  
объяснять что такое триггер и как его создавать, в каких случаях они нужны

Домашние задания

### 1 SP и function

1) Написать функцию возвращающую Клиента с наибольшей суммой покупки.

2) Написать хранимую процедуру с входящим параметром CustomerID, выводящую сумму покупки по этому клиенту.

Использовать таблицы :

Sales.Customers

Sales.Invoices

Sales.InvoiceLines

3) Создать одинаковую функцию и хранимую процедуру, посмотреть в чем разница в производительности и почему.

4) Создайте табличную функцию покажите как ее можно вызвать для каждой строки result set'a без использования цикла.

Во всех процедурах, в описании укажите для преподавателям

5) какой уровень изоляции нужен и почему.

Опционально

6) Переписываем одну и ту же процедуру kitchen sink с множеством входных параметров по поиску в заказах на динамический SQL.

Сравниваем планы запроса.

7) Напишите запрос в транзакции где есть выборка, вставка\добавление\удаление данных и параллельно запускаем выборку данных в разных уровнях изоляции, нужно предоставить мини отчет, что на каком уровне было видно со скриншотами и ваши выводы (1-2 предложение)

8) Сделайте параллельно в 2х окнах добавление данных в одну таблицу с разным уровнем изоляции, изменение данных в одной таблице, изменение одной и той же строки. Что в итоге получилось, что нового узнали.

---

## 14 Создание и использование CLR

создавать процедуры с использованием CLR

Домашние задания

### 1 Создаем CLR

Варианты ДЗ (сделать любой один):

1) Взять готовую dll, подключить ее и продемонстрировать использование.  
Например, <https://sqlsharp.com>

2) Взять готовые исходники из какой-нибудь статьи, скомпилировать, подключить dll, продемонстрировать использование.  
Например,  
<https://www.sqlservercentral.com/articles/xlsexport-a-clr-procedure-to-export-proc-results-to-excel>

<https://www.mssqltips.com/sqlservertip/1344/clr-string-sort-function-in-sql-server/>

3) Написать полностью свое (что-то одно):  
\* Тип: JSON с валидацией, IP / MAC - адреса, ...  
\* Функция: работа с JSON, ...  
\* Агрегат: аналог STRING\_AGG, ...  
\* (любой ваш вариант)

Результат ДЗ - исходники (\*), скрипт подключения dll, демонстрация использования.

## 15 Уровни изоляции транзакций

объяснять разницу в уровнях изоляции транзакций, проходить тест по тому какой уровень когда нужно применять  
объяснять какие блокировки каждый уровень накладывает  
создавать процедуры с обработкой ошибок на SQL

## 2 Проектирование БД

### 1 Проектирование БД

проектировать БД  
нормализовывать таблицы

Домашние задания

#### 1 Начало проекта

Смотрим на схему, которая получилась в создании таблиц через DDL

Строим схему в use-case средстве, генерируем SQL код для создания.

Приводим свою БД в 3ю нормальную форму, либо если таблица в 1нф или 2нф или вообще не нормализована пишем зачем и почему так лучше.

В качестве проекта вы можете взять любую идею, которая вам близка и сделать схему базы данных, а затем создать ее.

Это может быть какая-то часть вашего рабочего проекта, которую вы хотите переосмыслить.

Если есть идея, но не понятно как ее уложить в рамки учебного проекта, напишите преподавателю и мы поможем.

На занятии-семинаре по представлению проектов в конце 2го модуля нужно будет показать схему БД, объяснить бизнес идею, а также технические решения, которые вы считаете важными.

### 2 Подходы к проектированию БД

проектировать БД исходя из требований  
учитывать аспекты безопасности при проектировании БД

---

<b>3 Индексы</b>	создавать индексы определять какие индексы не нужны  Домашние задания  1 Какие индексы вам нужны  Думаем какие запросы у вас будут в базе и добавляем для них индексы. Проверяем, что они используются в запросе.
<b>4 Проектируем БД для OLTP нагрузки. Почему важно разделить OLTP и OLAP.</b>	различать OLTP и OLAP проектировать БД для OLTP нагрузки
<b>5 Проектирование хранилищ данных и OLAP систем</b>	проектировать хранилища данных  Домашние задания  1 Домашнее задание  Создать куб в рамках своего проекта/решения
<b>6 Семинар по созданным проектам БД</b>	обсудить и получить обратную связь по проекту

# 3 Оптимизация и Администрирование БД

- 1 Планы запросов, а теперь подробнее.**

анализировать план запроса и делать выводы как проводить оптимизацию  
интерпретировать статистики по времени и вводу выводу  
определять parameter sniffing и устранять его

---
- 2 DMV в SQL Server**

различать разные DMV  
использовать DMV для работы

---
- 3 Оптимизация работы базы данных**

находить проблемные места в работе сервера БД  
находить и устранять запросы, которые нагружают сервер БД  
интерпретировать информацию из Activity Monitor и делать выводы о характере нагрузки  
использовать профайлер для нахождения ресурсоемких запросов  
использовать механизм extended events

---
- 4 Популярные Hint'ы и подсказки оптимизатору**

перечислить в каких случаях какие хинты нужно и можно применять  
понимать минусы и плюсы использования хинтов и принимать решения о целесообразности их использования

## Домашние задания

### 1 Оптимизируем запрос

Цель: Используем все свои полученные знания для оптимизации сложного запроса.

Вариант 1.

Вы можете взять запрос со своей работы с планом и показать, что было до оптимизации, какие решения вы применили, и что стало

после.

В этом случае нужно приложить Текст запроса, актуальный план и статистики по времени и операциям ввода\вывода до оптимизации и после оптимизации.

Опишите кратко ход рассуждений при оптимизации.

Вариант 2.

Оптимизируйте запрос по БД WorldWideImporters.

Приложите текст запроса со статистиками по времени и операциям ввода вывода, опишите кратко ход рассуждений при оптимизации.

```
Select ord.CustomerID, det.StockItemID,
SUM(det.UnitPrice), SUM(det.Quantity),
COUNT(ord.OrderID)
FROM Sales.Orders AS ord
JOIN Sales.OrderLines AS det
ON det.OrderID = ord.OrderID
JOIN Sales.Invoices AS Inv
ON Inv.OrderID = ord.OrderID
JOIN Sales.CustomerTransactions AS Trans
ON Trans.InvoiceID = Inv.InvoiceID
JOIN Warehouse.StockItemTransactions AS
ItemTrans
ON ItemTrans.StockItemID = det.StockItemID
WHERE Inv.BillToCustomerID != ord.CustomerID
AND (Select SupplierId
FROM Warehouse.StockItems AS It
Where It.StockItemID = det.StockItemID) = 12
AND (SELECT
SUM(Total.UnitPrice*Total.Quantity)
FROM Sales.OrderLines AS Total
Join Sales.Orders AS ordTotal
On ordTotal.OrderID = Total.OrderID
WHERE ordTotal.CustomerID = Inv.CustomerID)
> 250000
AND DATEDIFF(dd, Inv.InvoiceDate,
ord.OrderDate) = 0
GROUP BY ord.CustomerID, det.StockItemID
```

Используем DMV, хинты и все прочее для сложных случаев

---

5 **Семинар: Оптимизация сложных запросов**

оптимизировать запрос  
анализировать план запроса  
анализировать и принимать решения по оптимизации статистики ввода вывода по запросу

---

6 **Еще раз об индексах**

обновлять статистику  
создавать недостающие индексы  
находить и удалять неиспользующиеся индексы  
обслуживать индексы

---

7 **OLAP часть 2**

изучить как работать с ssis  
проектировать по Data Vault

---

8 **Очереди в MS SQL Server**

создавать очереди в SQL Server

Домашние задания

1 Создание очереди

Создание очереди в БД для фоновой обработки задачи в БД.

---

9 **Секционирование таблиц** определять когда нужно секционировать таблицы секционировать таблицы

Домашние задания

1 Секционирование таблицы

Выбираем в своем проекте таблицу-кандидат для секционирования и добавляем партиционирование.

Если в проекте нет такой таблицы, то делаем анализ базы данных из первого модуля, выбираем таблицу и делаем ее секционирование, с переносом данных по секциям (партициям) - исходя из того, что таблица большая, пишем скрипты миграции в секционированную таблицу

---

10 **Резервное копирование и восстановление** создавать backup базы данных делать восстановление из бэкапа

---

11 **Репликация данных, Job'ы и SQL Agent** настраивать репликацию выбирать нужный вид репликации

---

12 **Безопасность в SQL Server** Немного про пользователей и политики безопасности в SQL Server.

---

## 13 Обсуждение проектов. Подводим итоги курса.

презентовать свой проект  
получить обратную связь  
принять поздравления об окончании курса

### Домашние задания

#### 1 Защита проектов

На демонстрации нужно будет рассказать про ваш проект:

- 1) описание бизнес идеи, какие проблемы решаются
- 2) требования к проекту по производительности, функциональности, безопасности
- 3) показать схему базы данных
- 4) созданную базу данных с данными - хотя бы пару строк в основных таблицах
- 5) продемонстрировать работу БД по основному бизнес потоку, например, если вы сделали интернет магазин, то нужно показать как пользователь будет делать заказ - запустить процедуры или SQL, который сформирует корзину, добавит товары на доставку и т.д.
- 6) рассказать какие решения были приняты вам, что соответствовать требованиям к проекту, например, на какую нагрузку вы рассчитываете и какие решения приняли, чтобы ее выдержать.
- 7) принять поздравления о завершении программы и защите проекта.