Я не чувствую себя достаточно уверенно с SQL (в особенности с MS SQL), поэтому я постараюсь расписать всё что я думал при выполнении заданий и, если что-то не так, надеюсь что вы мне поможете.

Так как я не совсем уверен в том, что означает каждый столбец в таблице (хоть я и построил диаграмму бд), я воспользовался вот [этим](https://dataedo.com/download/AdventureWorks.pdf) документом чтобы убедиться в назначении колонок в таблицах.

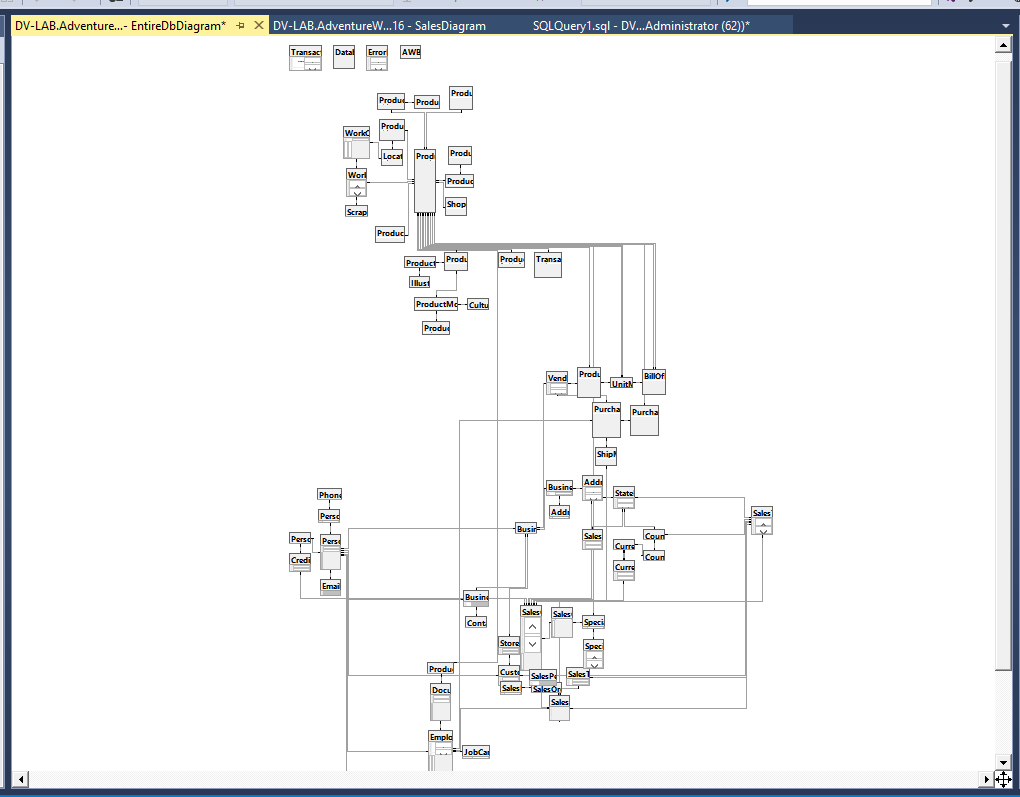
# Задачи:

Задача 1.

Скачать БД AdventureWorks 2016 (OLTP) со страницы https://docs.microsoft.com/en-us/sql/samples/adventureworks-install-configure

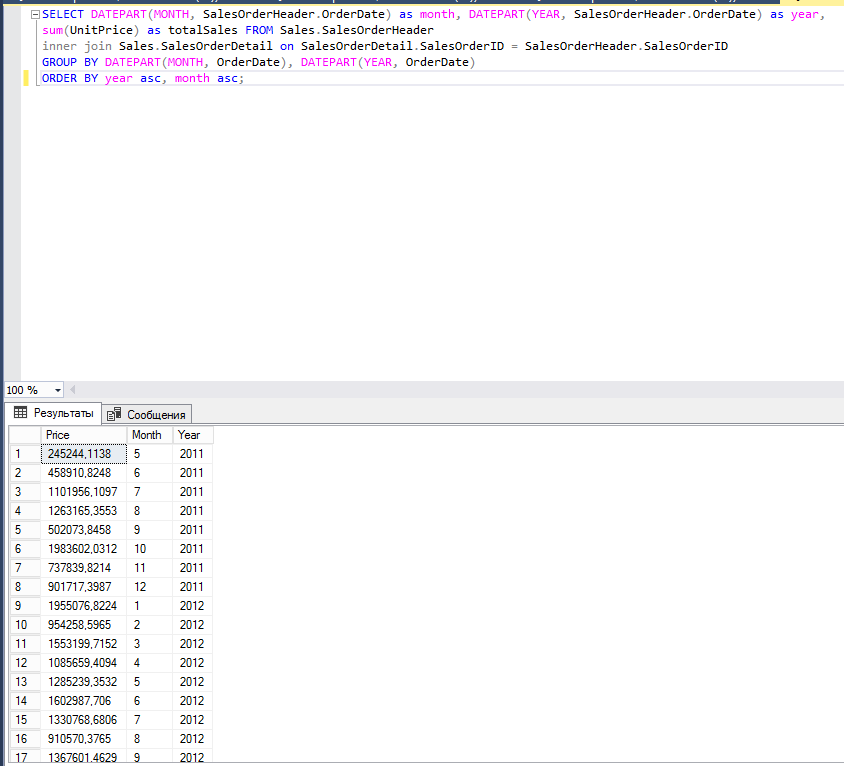
Восстановить бэкап на SQL сервер в виртуальной машине

Изучить схему БД AdventureWorks2016

Схему изучил, красиво: 

Задача 2

Вывести общую сумму продаж с разбивкой по месяцам, за все время работы компании



Задача 3

Выбрать 10 самых приоритетных городов для следующего магазина

Столбцы: Город | Приоритет

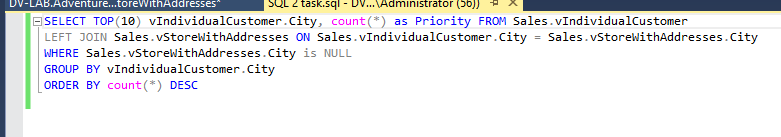
Приоритет определяется как количество покупателей в городе

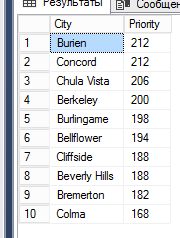
В городе не должно быть магазина

Я не до конца уверен откуда брать то, сколько покупателей в городе. Буду использовать представление Sales.vIndividualCustomer для этих целей.

Информацию о том, в каком городе магазины, буду брать из Sales.vStoreWithAddresses

При помощи LEFT JOIN я выбрал всех клиентов и оставил только тех, у которых нет в городе магазина при помощи WHERE, сгруппировал их по городу и подсчитал их количество.





Задача 4

Выбрать покупателей, купивших больше 5 единиц одного и того же продукта

Столбцы: Фамилия покупателя | Имя покупателя | Название продукта | Количество купленных экземпляров (за все время)

Упорядочить по имени покупателя по возрастанию, затем по количеству купленных экземпляров по убыванию

Изначально надо было выяснить как получить имя и фамилию заказчика. Если я правильно понял, то Customer.PersonID это Person.BusinessEntityID потому что не у каждого Customer.CustomerID есть PersonID, из-за чего не получится, как я понял, получить имя и фамилию этого заказчика

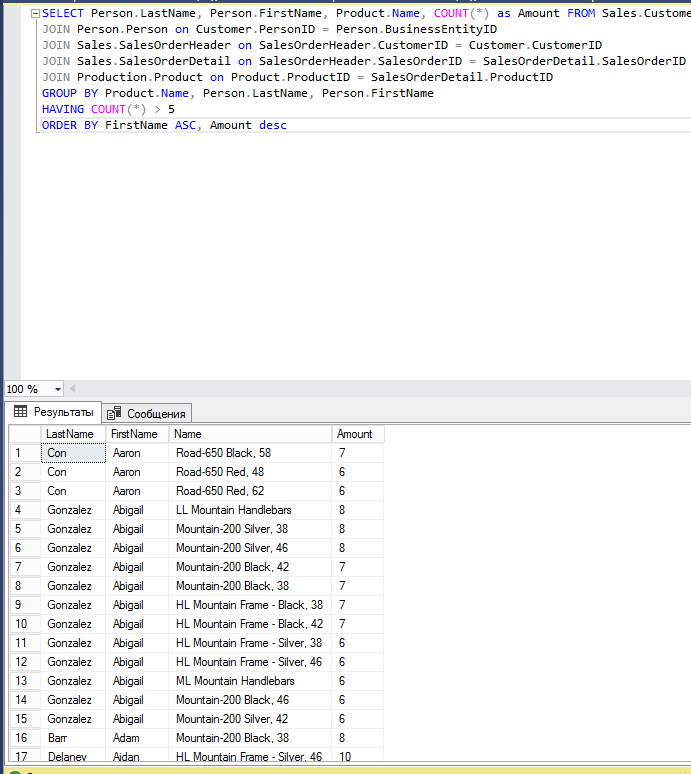
Получается что:

Первый джойн для получения имени и фамилии

Второй для получения заказов для Person

Третий для получения информации о том, какой продукт был заказан

Четвёртый для получения названия продукта



Задача 5

Вывести содержимое первого заказа каждого клиента

Столбцы: Дата заказа | Фамилия покупателя | Имя покупателя | Содержимое заказа

Упорядочить по дате заказа от новых к старым

В столбец содержимого заказа нужно объединить все элементы заказа в следующем формате:

<Имя товара> Количество: <количество в заказе> шт.

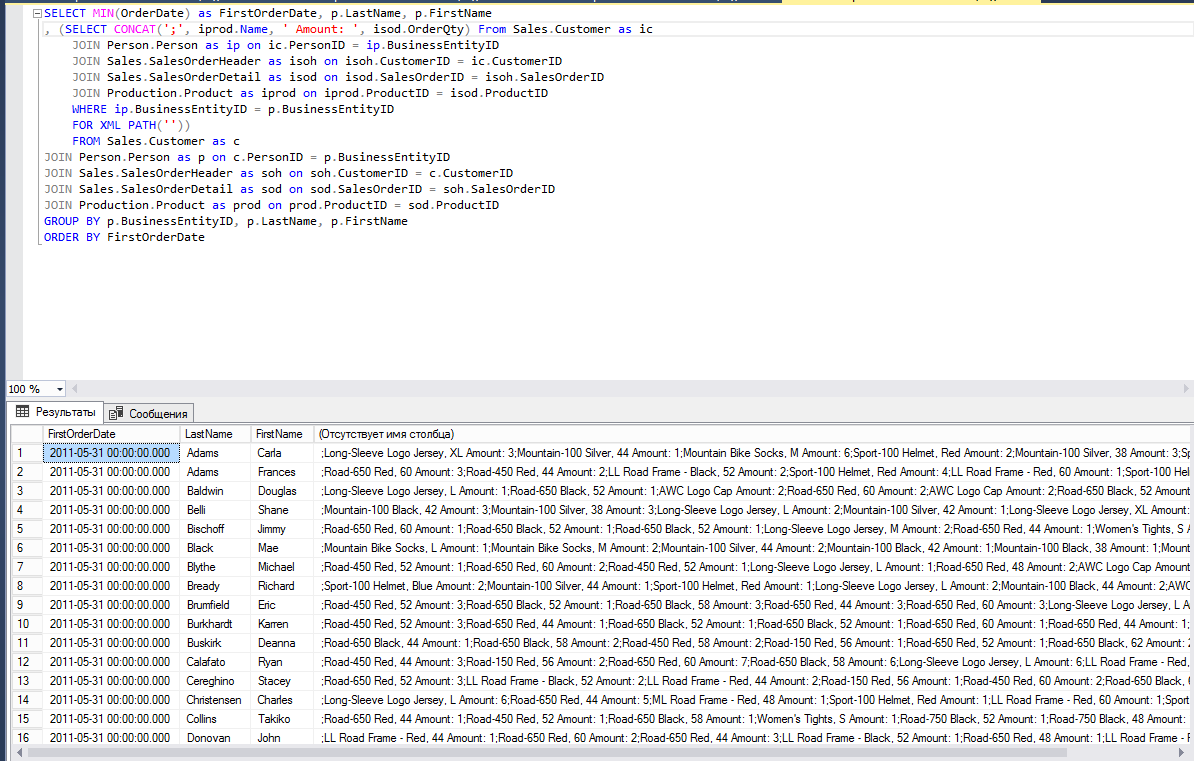
<Имя товара> Количество: <количество в заказе> шт.

<Имя товара> Количество: <количество в заказе> шт.

...

Из-за того что у нас версия БД 2016, мне пришлось делать внутренний почти полностью идентичный запрос. Мне кажется, что это сильно больше чем нужно для выполнения этого задания. Во всяком случае, в 2017 версии можно было бы сделать STRING\_AGG().

При этом я так же не смог убрать первую ; тк функция STUFF, по непонятной мне причине убирает ВСЕ ;



Ещё, у меня collation не воспринимает кирилицу, поэтому результат на английском. Я пытался поменять collation у бд, но получаю ошибку во многих местах. Как я выяснил, это из-за привязки

Задача 6

Вывести содержимое сотрудников, непосредственный руководитель которых младше и меньше работает в компании

Столбцы: Имя руководителя | Дата приема руководителя на работу| Дата рождения руководителя |

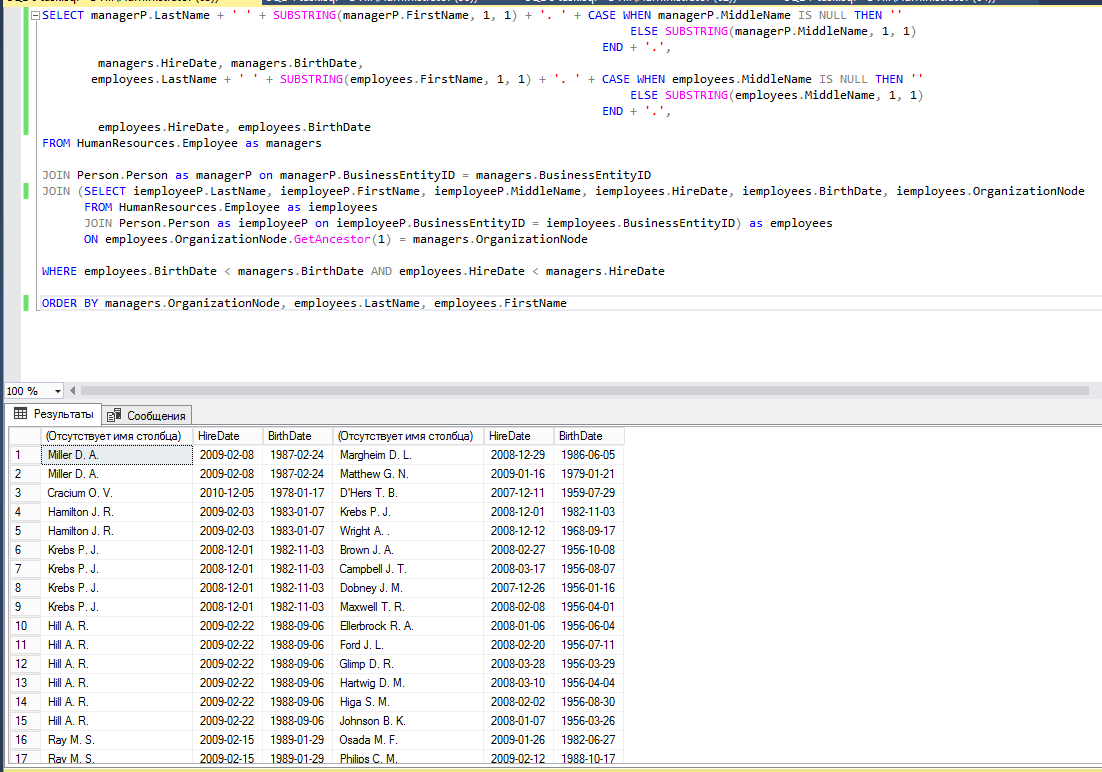
| Имя сотрудника | Дата приема сотрудника на работу| Дата рождения сотрудника

Поле имя выводит в формате 'Фамилия И.О.'

Упорядочить по руководителю от директора вниз к сотрудникам

Внутри одного уровня иерархии упорядочить по фамилии и имени сотрудника

Долго пытался понять что за странный OrganizationNode, но потом выяснил что это hierarchyid - структура данных специфичная для MS SQL и решил что просто стоит сделать JOIN работников старше своих руководителей и работающих дольше них.



Задача 7

Написать хранимую процедуру, с тремя параметрами

входной параметр - две даты, с и по

выходной параметр - количество найденных записей

Результирующий набор содержит записи всех холостых мужчин-сотрудников, родившихся в диапазон указанных дат

Если я правильно понял, выходной набор - просто запрос исполняемый внутри хранимой процедуры. В таком случае имеем следующее

