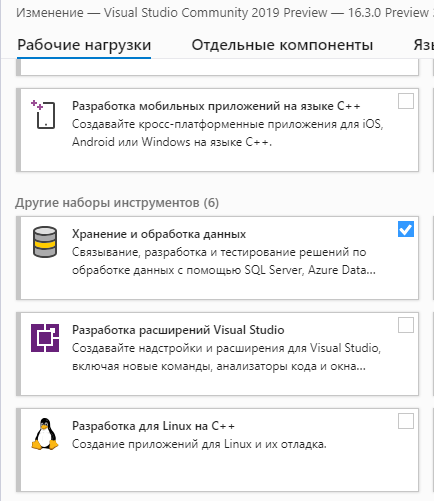
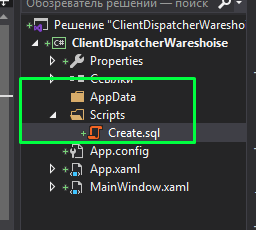
**Как создать и работать с СУБД MSSQL ?**

1. Требуется установить Visual Studio – можно конечно и без него обойтись, а использовать обычный Manager Studio, но сейчас наша задача продемонстрировать работу с СУБД создав GUI приложение.
2. В инсталяторе Visual Studio выбрать для установки компонент «Хранение и обработка данных» и нажать кнопку «установить».



1. Открыть Visual Studio и создать проект «WPF .Net Framework»
2. После создания проекта и указав ей какое-либо имя (в моем случаи это «ClientDispatcherWareshoise») – нужно добавить 2 папки: «AppData» и «Scripts».



1. В папку «Scripts» - нужно добавить файл с скриптом создания БД. В моем случаи я назвал «Create.sql». (Расширение .sql – обязательно должно быть таким).
2. В этом файле нужно прописать скрипт, который будет создавать автоматически БД на вашей локальной машине. В соответствии со своей темой работы мой скрипт представлен следующим образом:

create database Wareshoise;

GO

use Wareshoise;

GO

CREATE TABLE plants

(

id\_plants INT PRIMARY KEY IDENTITY(1,1),

plant\_name varchar(40),

phone\_number varchar(15)

);

GO

CREATE TABLE goods

(

id\_goods INT PRIMARY KEY IDENTITY(1,1),

good\_name varchar(40),

number INT

);

GO

CREATE TABLE organizations

(

id\_organizations INT PRIMARY KEY IDENTITY(1,1),

organization\_name varchar(40),

phone\_number varchar(15)

);

GO

CREATE TABLE orders\_import

(

id\_import\_order INT PRIMARY KEY IDENTITY(1,1),

order\_date DateTime,

order\_end\_date DateTime,

price INT,

count INT,

ready BIT,

id\_goods INT,

id\_plants INT,

FOREIGN KEY (id\_goods) REFERENCES goods (id\_goods),

FOREIGN KEY (id\_plants) REFERENCES plants (id\_plants)

);

GO

CREATE TABLE orders\_export

(

id\_export\_order INT PRIMARY KEY IDENTITY(1,1),

order\_date DateTime,

order\_end\_date DateTime,

price INT,

count INT,

ready BIT,

id\_goods INT,

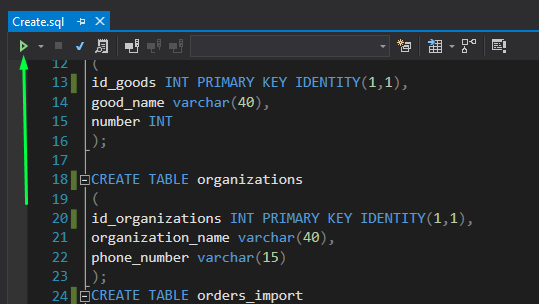
id\_organizations INT,

FOREIGN KEY (id\_goods) REFERENCES goods (id\_goods),

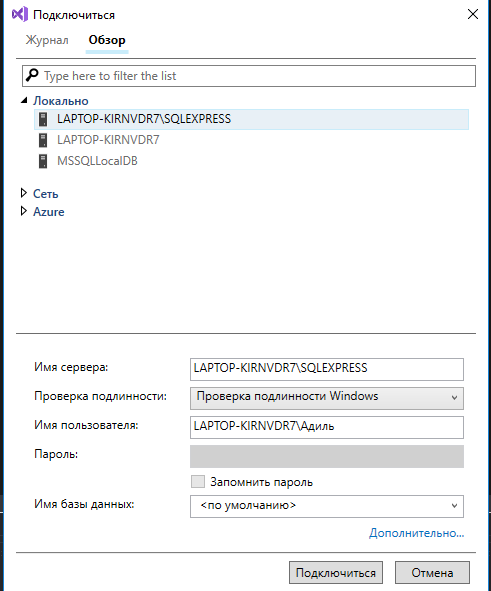
FOREIGN KEY (id\_organizations) REFERENCES organizations(id\_organizations)

);

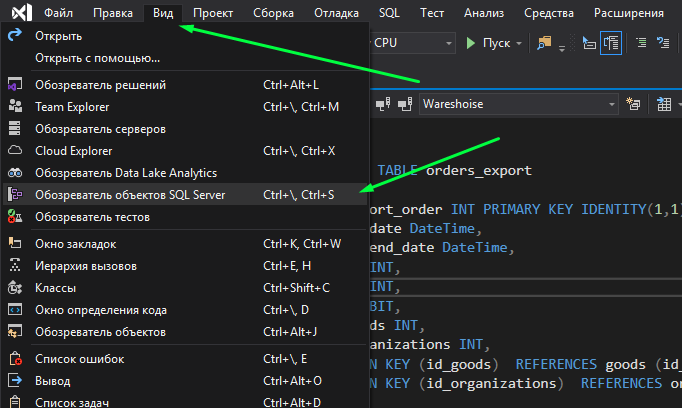
1. Создав файл со скриптом, можно его использовать для генерации БД



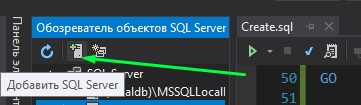
1. Далее вам будут предложены варианты то, как вы хотите развернуть БД. Здесь для ознакомительных целей – выбираем «Локально->» и далее рекомендуется выбрать имя сервера, который заканчивается на SQLEXPRESS, ну а если этого варианта нет, то можете выбрать любой из оставшихся, после чего нажать «Подключиться»



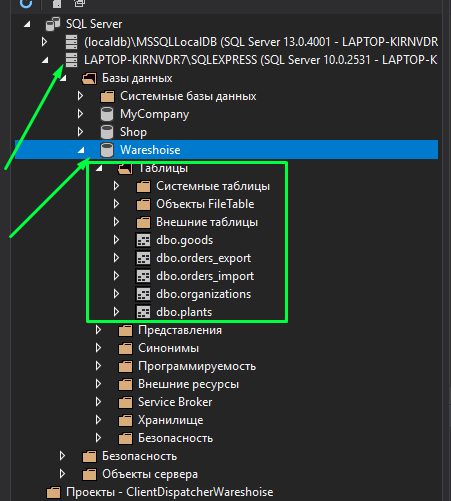
1. Далее в нижнем панели вам должны сообщить ваши ошибки/успех выполнения скрипта. Для проверки, что БД действительно создалась – выполняем следующие шаги:
   1. Открываем «Обозреватель объектов SQL Server»



* 1. Жмем в появившемся окне «Добавить SQL Server»



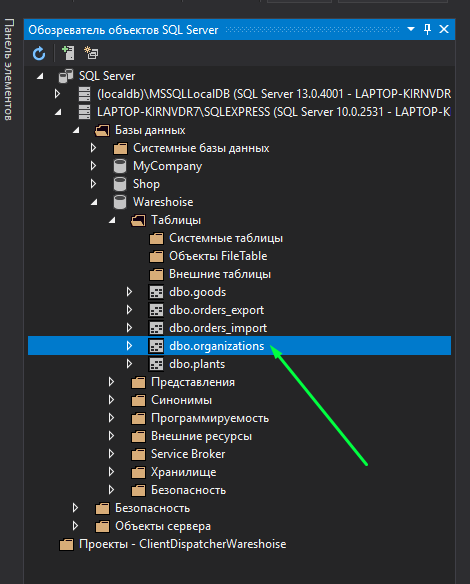
* 1. В появившемся окне снова подключаемся к серверу, который выбирали для генерации БД в п. 8
  2. В списке «Обозреватель объектов SQL Server» после этих операции должен был появиться sql сервер. Выбрав его мы должны заметить, что БД действительно сформировалось



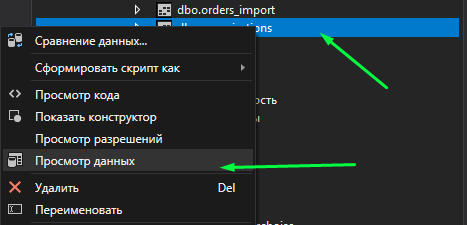
Отлично, БД мы создали, но теперь требуется добавить в них данные.

Здесь можно пойти 3-мя путями:

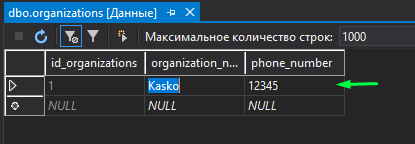
1. **Используя GUI:**
   1. Выбираем таблицу, в которую хотим добавить данные



* 1. Жмем по ней ПКМ и выбираем «Просмотр данных»

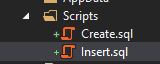


* 1. Вводим нужные данные и жмем «Enter»



Все, данные добавились.

1. **Используя язык SQL/T-SQL:**
   1. Добавляем новый скрипт в папку «Scripts» с названием «Insert.sql»



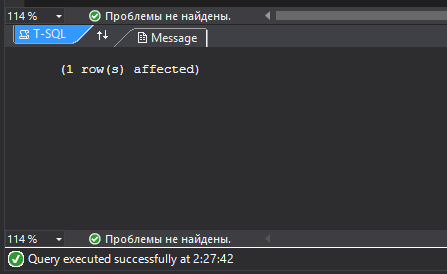
* 1. Пишем скрипт для добавления новой информации в таблицу

USE Wareshoise;

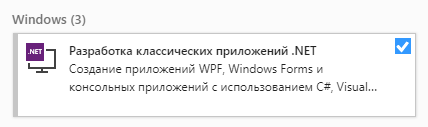
GO

INSERT INTO [dbo].[organizations] (organization\_name, phone\_number) VALUES ('MIREA', 88005553535);

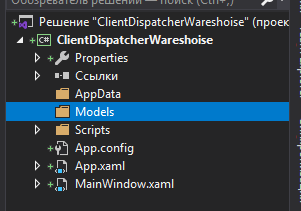
* 1. Ниже должно высветится сообщение об успешном sql запросе



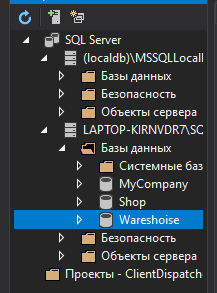
1. **Использование ORM –** один из самых простых способов, который не требует от разработчика знаний в языке SQL и решает эту проблему знаниями ООП:
2. Заходим снова в инсталятор Visual Studio и устанавливаем новый компонент



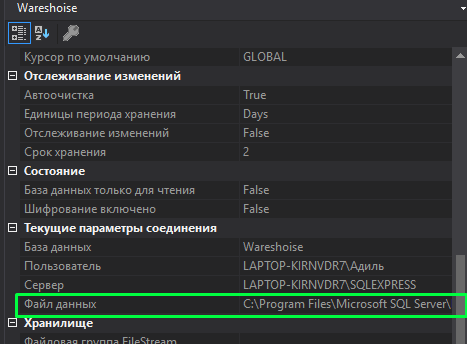
1. После установки возвращаемся к нашему проекту и добавляем новую папку «Models», где будет хранится модель нашей БД



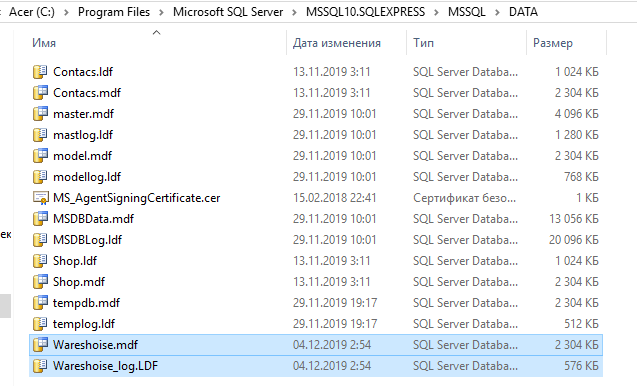
1. Копируем файл с БД, который создали
   1. Жмем на файл с БД



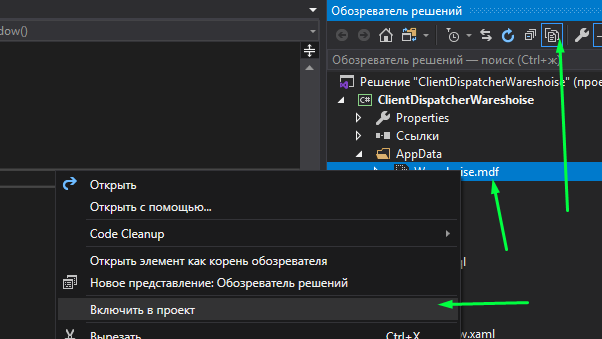
* 1. Справа высветятся свойства файла с БД, где копируем и переходим по заданному пути



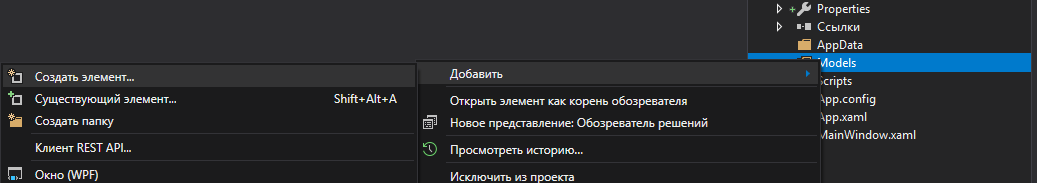
* 1. В этой папке копируем нужные файлы в папку «AppData» нашего проекта



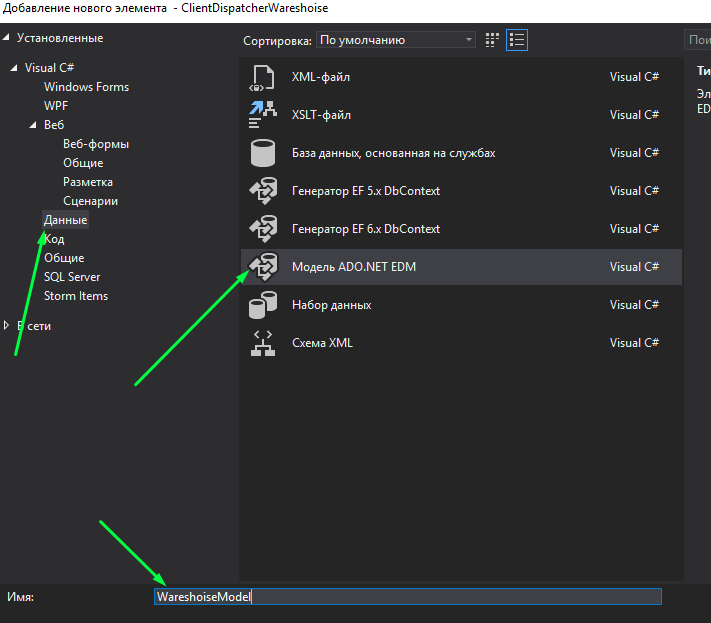
1. Включаем БД в проект



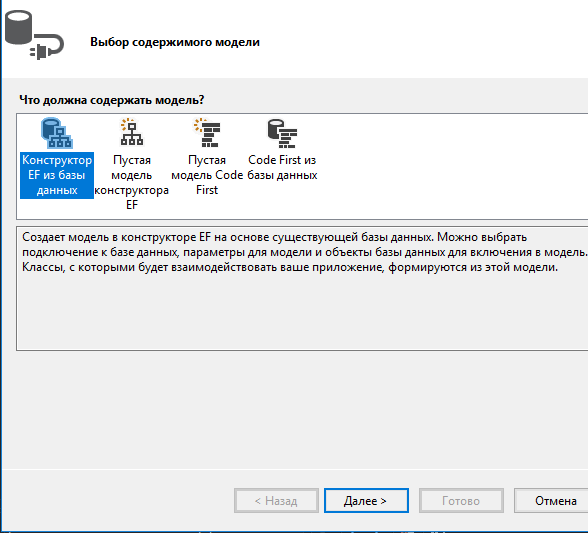
1. Создаем новый элемент в папке «Models»



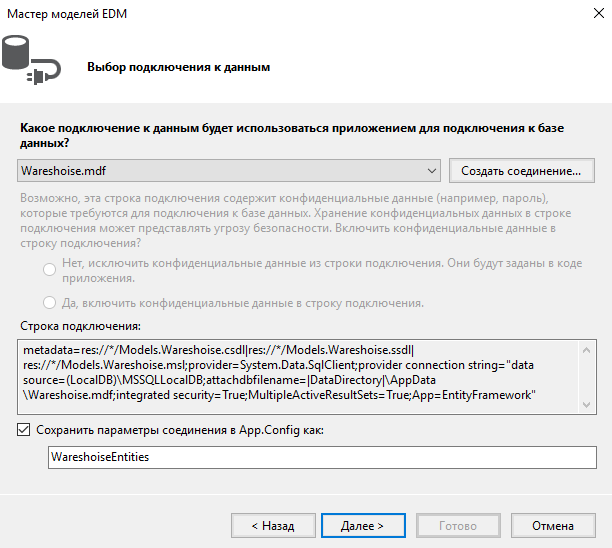
1. Выбираем следующий элемент и устанавливаем имя модели



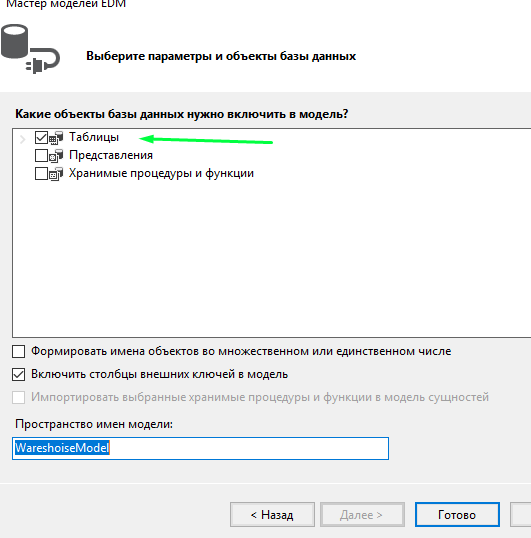
1. Жмем далее



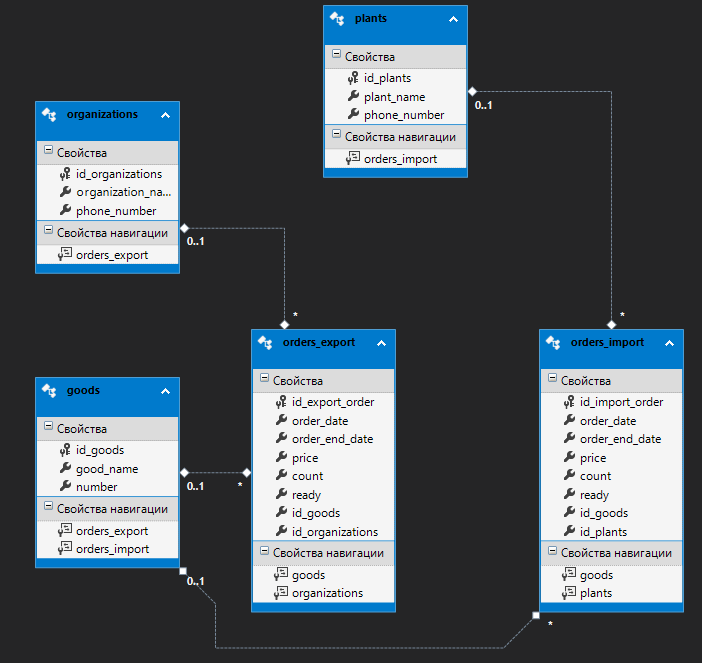
1. Выбираем наш файл с БД и жмем далее



1. Жмем далее до появления следующего окна. Тут нужно поставить галочку «Таблицы», чтобы выбрать все таблицы из БД. Жмем «Готово».

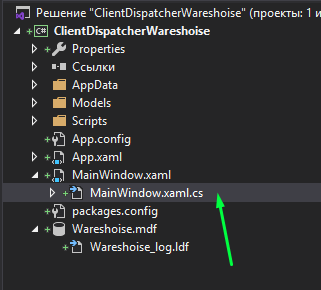


1. Потом вам будут выходить всплывающие сообщения, где всегда нужно нажимать «ок»
2. Далее у вас появится окно, где вы сможете полюбоваться на сформированную модель БД

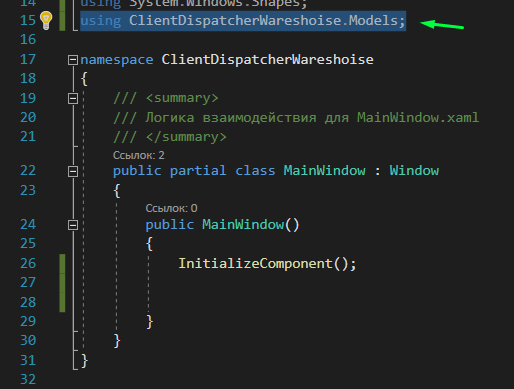


Теперь собственно о том как добавить данные. (Следующий пример будет лишь демонстрацией того, что может делать ORM, но не в коем случаи не рекомендуется писать данный код в том же месте)

1. Переходим в MainWindow.xaml.cs



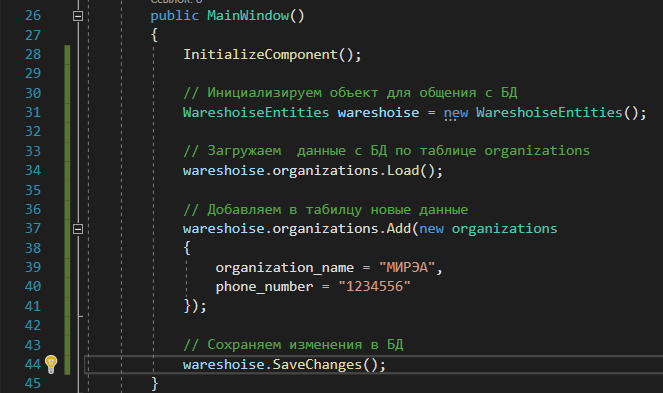
1. Подключаем пространство имен, где лежит сформированная модель БД



1. Подключаем ещё следующие для упрощения работы с БД



1. Дальше осталось написать код



Все изменения сохранились в таблице после запуска приложения и как можно заметить – здесь мы не использовали знания SQL совсем.