**DatabaseRestaurant**

-------------------------------------------------------------------------------------------------

Отваряме SQL Management Studio.

Първо правим настройката Tools->Options->Designers -> Махаме Prevent savingh changes that …….

Създаваме база данни RestaurantsT.

Изъплняваме последователно заявките от Cities.sql, Categories.sql, Restaurants.sql (намират се в DatabaseRestaurants \Restaurants.DB\dbo\Tables) за създаване на таблиците Cities, Categories, Restaurants.

----------------------------------------------------------------------------------------------------------------------------

Създаваме Blank Solution

VS2015 -> File->New Project Search: Blank Solution -> DatabaseRestaurants

Добавяме проект -> Add New Project -> SQl Server -> SQL Server Database Project с името „Restaurants.DB“.

---

SQL проекта е полезен когато искаме да разгледаме структурата на базата данни. Също така чрез такъв проект можем да експортнем базата данни във вид на SQL заявки които я създават.

----

Избираме контекстното меню на проекта Restaurants.DB и оттам SchemaCompare…

Имаме две текстови кутии. Лявата е нашия проект, а във втората трябва да посочим базата данни: Select Target -> Database -> Select Connection ->Show Connection Properties:

//името на сървъра когато е локален може да бъде “.“ или „.\SQLEXPRESS“

//Точката „.“ замества името „localhost“

Server Name: .\SQLEXPRESS

Database Name: RestaurantsT

-> Connect

Може от SqlSchemacompare да изберем Switch source and target -> после Compare

Селектираме таблиците и даваме Update. Така ще ни се генерират sql скриптове за създаване на таблиците в проекта в директорията dbo.

---------

Важно е на Restaurants.DB/Properties да посочим sql сървъра, който се използва -> Target platform -> и оттам да посочим правилната версия на използвания SQL Server.

В противен случай няма да може да се Publish.

----------------

Добавяме към директорията dbo -> Add New Item -> User Script -> Post Deployment (в него си копираме всичко от съществуващ проект от Script.PostDeployment1.sql)

Този скрипт може да го изпълним в началото за да си инициализираме базата с някакви данни. Това става чрез

Контексно меню на Restaurants.DB -> Publish…. -> Target database connection: RestaurantsT -> Publish

-----------------

За връзка на едно приложение (C#) към базата данни е необходим Connection String. Този стринг ни дава информация за това къде се намира сървъра (локално, по мрежата), коя база данни искаме да ползваме и как да се автентикираме (потребител или с локална автентикация).

Connection String може да се добави чрез Server Explorer.

Отваряме Server Explorer от VS2015

Избираме Connect to database -> Data Source -> MS SQL Server

Server name: .\SQLEXPRESS -> Refresh

Authentication: Windows Authentication

Select or enter a database name: RestaurantsT

-> Connect

След като се свържем към дадената база данни, може да дадем десен бутон -> Properties и оттам ще може да си вземем connection string, който се намира в поле с такова име.

-------------------------------------------------

Ще ни трябва един нов проект, който да съдържа информация за това с какви обекти можем да работим (Entities) и каква функционалност да имат (Repositories). Тъй като класовете от този проект ще бъдат използвани на други места (конзолно приложение, уеб приложение и др.) той може да бъде от тип библиотека (без да има изпълними файлове).

Създаваме в нашия solution -> проект (Class library) с името „DataAccess“ и 2 папки - Entities и Repositories.

Създаване на 3 class-a в Entities - Restaurant, Category и City със структура подобна на таблиците в базата.

Обяснение на идеята да се ползва Repository - еднотипна CRUD фунционалност.

-> виж шаблона за дизайн Repository

Осъзнаване на необходимостта за базов клас BaseRepository. Тъй като на всяко repository ще му трябва ConnectionStrin, те могат да го наследят като пропърти от базовия клас където е дефинирано.

Добвяме и останалите repository класове.

---------------------------------------

Създаваме console application с името „Restaurants.Console“ за демонстрация.

Друго което ни трябва е да достъпим DataAccess-a

Restaurants.Console -> References –> Add Reference

И маркираме DataAccess.

Чрез (Ctrl+.) на нереферираните класове добавяме необходимите using.

Тъй като преди това стартирахме прокта SQL Database project е необходимо да направим нашето конзолно приложение да се стартита по подразбиране.

Десен бутон -> Set As StartUp Project.

---------------------------------------------------------------------------

Създаваме съхранена процедура:

Отиваме в SQL Management Studio, създаваме заявка към базата данни Restaurants и изпълняваме заявката по създаване на процедурата:

create PROCEDURE DeleteRestaurant

(

@id int

)

AS

BEGIN

DELETE FROM Restaurants

WHERE ID = @id;

END

След това се връщаме във VS2015 и отваряме проекта Restaurants.DB, от там избираме Schema compare… Отваря се SqlSchemacompare и от там посочваме в дясното падащо меню базата данни Restaurants. След това Switch source and target -> после Compare.

Виждаме че има добвена нова промяна – това е съхранената процедура. Избираме Update за да може да се добави към проекта Restaurants.DB.

Добавяме метода DeleteByID() към RestaurantRepository в проекта DataAccess/Repositories.

Пробваме новия метод в проекта Restaurants.Console.