­­МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

­­­

Лабораторна робота №1

з курсу «**Комп’ютерні технології проектування**»

для студентів базового напрямку 6.08.04 "Комп’ютерні науки"

(заочна форма навчання)

Варіант 14

Виконав студент гр. КНз-3

Чалий Михайло

­­

Львів 2014

## Мета роботи

Мета роботи – ознайомитись з структурованою мовою запитів SQL та з середовищем розробки баз даних Microsoft SQL Server 2008, здобути навики роботи в IDE Visual Studio 2010 з використанням об’єктно-реляційної проекції Entity Framework.

## Теоретичні відомості

SQL (англ. Structured query language — мова структурованих запитів) — декларативна мова програмування для взаємодії користувача з базами даних, що застосовується для формування запитів, оновлення і керування реляційними БД, створення схеми бази даних і її модифікації, системи контролю за доступом до бази даних. Сам по собі SQL не є ні системою керування базами даних, ні окремим програмним продуктом. Не будучи мовою програмування в тому розумінні, як C або Pascal, SQL може формувати інтерактивні запити або, будучи вбудованою в прикладні програми, виступати в якості інструкцій для керування даними. Окрім того, стандарт SQL вміщує функції для визначення зміни, перевірки і захисту даних.

## Завдання

1. Створити БД згідно з індивідуальним завданням.
2. Згенерувати скрипт створення БД.
3. Написати скрипти для заповнення БД тестовою інформацією.
4. Створити новий проект в IDE Visual Studio 2010, в ньому створити ORM створеної в п.1 БД.
5. Створити проекти типу Console Application і в ньому продемонструвати приклад роботи зі створеною ORM.
6. Всі написані та згенеровані скрипти внести в систему контролю версій.

## Реалізація

Лістінг Program.cs

namespace Restaurants

{

using ManyConsole;

using System;

using System.Data.Entity;

using System.Linq;

using System.Linq.Expressions;

class Program

{

static int Main(string[] args)

{

var commands = ConsoleCommandDispatcher.FindCommandsInSameAssemblyAs(typeof(Program));

return ConsoleCommandDispatcher.DispatchCommand(commands, args, Console.Out);

}

public class Add : ConsoleCommand

{

public Add()

{

IsCommand("add", "Додати тестовi даннi");

}

public override int Run(string[] remainingArguments)

{

using (var db = new RestaurantEntities())

{

var ua = CreateIfNotExists(db.Countries, \_ => \_.Name == "Україна", () => new Country

{

Name = "Україна",

ISOCode = "UA"

}, "Країна Україна");

var uk = CreateIfNotExists(db.Countries, \_ => \_.Name == "Велика Британiя", () => new Country

{

Name = "Велика Британiя",

ISOCode = "UK"

}, "Країна Велика Британiя");

var lviv = CreateIfNotExists(db.Cities, \_ => \_.Name == "Львiв", () => new City

{

Name = "Львiв",

Country = ua,

}, "Мiсто Львiв");

var fest = CreateIfNotExists(db.RestaurantNetworks, \_ => \_.Name == "FEST", () => new RestaurantNetwork

{

Name = "FEST",

Country = ua

}, "Мережа FEST");

var kopalnia = CreateIfNotExists(db.Restaurants, \_ => \_.Name == "Копальня кави", () => new Restaurant

{

Name = "Копальня кави",

RestaurantNetwork = fest,

Address = "Площа ринок",

City = lviv

}, "Ресторан Копальня кави");

var miasoSparv = CreateIfNotExists(db.Restaurants, \_ => \_.Name == "М'ясо i справедливiсть", () => new Restaurant

{

Name = "М'ясо i справедливiсть",

RestaurantNetwork = fest,

Address = "Площа Данили Галицького",

City = lviv

}, "Ресторан М'ясо i справедливiсть");

db.SaveChanges();

}

Console.WriteLine("Тестовi даннi створенi...");

return 0;

}

private static T CreateIfNotExists<T>(DbSet<T> set,

Expression<Func<T, bool>> predicate,

Func<T> factory, string description)

where T : class

{

var entity = set.FirstOrDefault(predicate);

if (entity == null)

{

entity = set.Add(factory());

Console.WriteLine("Створено сутнiсть " + description);

}

else

{

Console.WriteLine("Сутнiсть " + description + " вже iснує");

}

return entity;

}

}

public class View : ConsoleCommand

{

public View()

{

IsCommand("view", "Дивитись даннi");

}

public override int Run(string[] remainingArguments)

{

using (var db = new RestaurantEntities())

{

Console.WriteLine("Ресторани Львова");

var lviv = db.Cities.First(\_ => \_.Name == "Львiв");

foreach (var restoraunt in lviv.Restaurants)

{

Console.WriteLine("Ресторан " + restoraunt.Name);

}

}

return 0;

}

}

}

}

Лістінг створення бази данних

USE [master]

GO

/\*\*\*\*\*\* Object: Database [Restaurant] Script Date: 19.10.2014 13:05:06 \*\*\*\*\*\*/

CREATE DATABASE [Restaurant]

CONTAINMENT = NONE

ON PRIMARY

( NAME = N'Restaurant', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL12.SQLEXPRESS\MSSQL\DATA\Restaurant.mdf' , SIZE = 5120KB , MAXSIZE = UNLIMITED, FILEGROWTH = 1024KB )

LOG ON

( NAME = N'Restaurant\_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL12.SQLEXPRESS\MSSQL\DATA\Restaurant\_log.ldf' , SIZE = 1024KB , MAXSIZE = 2048GB , FILEGROWTH = 10%)

GO

ALTER DATABASE [Restaurant] SET COMPATIBILITY\_LEVEL = 120

GO

USE [Restaurant]

GO

/\*\*\*\*\*\* Object: Table [dbo].[City] Script Date: 19.10.2014 13:05:06 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[City](

[CityID] [uniqueidentifier] ROWGUIDCOL NOT NULL CONSTRAINT [DF\_City\_CityID] DEFAULT (newid()),

[Name] [nvarchar](455) NOT NULL,

[CountryID] [uniqueidentifier] NOT NULL,

CONSTRAINT [PK\_City] PRIMARY KEY CLUSTERED

(

[CityID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Country] Script Date: 19.10.2014 13:05:06 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Country](

[CountryID] [uniqueidentifier] ROWGUIDCOL NOT NULL CONSTRAINT [DF\_Country\_CountryID] DEFAULT (newid()),

[Name] [nvarchar](455) NOT NULL,

[ISOCode] [nchar](10) NOT NULL,

CONSTRAINT [PK\_Country] PRIMARY KEY CLUSTERED

(

[CountryID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Restaurant] Script Date: 19.10.2014 13:05:06 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Restaurant](

[RestaurantID] [uniqueidentifier] ROWGUIDCOL NOT NULL CONSTRAINT [DF\_Restaurant\_RestaurantID] DEFAULT (newid()),

[Name] [nvarchar](455) NOT NULL,

[Address] [nvarchar](455) NOT NULL,

[RestaurantNetworkID] [uniqueidentifier] NOT NULL,

[CityID] [uniqueidentifier] NOT NULL,

CONSTRAINT [PK\_Restaurant] PRIMARY KEY CLUSTERED

(

[RestaurantID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[RestaurantNetwork] Script Date: 19.10.2014 13:05:06 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[RestaurantNetwork](

[RestaurantNetworkID] [uniqueidentifier] ROWGUIDCOL NOT NULL CONSTRAINT [DF\_RestaurantNetwork\_RestaurantNetworkID] DEFAULT (newid()),

[Name] [nvarchar](455) NOT NULL,

[CountryID] [uniqueidentifier] NOT NULL,

CONSTRAINT [PK\_RestaurantNetwork] PRIMARY KEY CLUSTERED

(

[RestaurantNetworkID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[City] WITH CHECK ADD CONSTRAINT [FK\_City\_Country] FOREIGN KEY([CountryID])

REFERENCES [dbo].[Country] ([CountryID])

GO

ALTER TABLE [dbo].[City] CHECK CONSTRAINT [FK\_City\_Country]

GO

ALTER TABLE [dbo].[Restaurant] WITH CHECK ADD CONSTRAINT [FK\_Restaurant\_City] FOREIGN KEY([CityID])

REFERENCES [dbo].[City] ([CityID])

GO

ALTER TABLE [dbo].[Restaurant] CHECK CONSTRAINT [FK\_Restaurant\_City]

GO

ALTER TABLE [dbo].[Restaurant] WITH CHECK ADD CONSTRAINT [FK\_Restaurant\_RestaurantNetwork] FOREIGN KEY([RestaurantNetworkID])

REFERENCES [dbo].[RestaurantNetwork] ([RestaurantNetworkID])

GO

ALTER TABLE [dbo].[Restaurant] CHECK CONSTRAINT [FK\_Restaurant\_RestaurantNetwork]

GO

ALTER TABLE [dbo].[RestaurantNetwork] WITH CHECK ADD CONSTRAINT [FK\_RestaurantNetwork\_Country] FOREIGN KEY([CountryID])

REFERENCES [dbo].[Country] ([CountryID])

GO

ALTER TABLE [dbo].[RestaurantNetwork] CHECK CONSTRAINT [FK\_RestaurantNetwork\_Country]

GO

USE [master]

GO

ALTER DATABASE [Restaurant] SET READ\_WRITE

GO

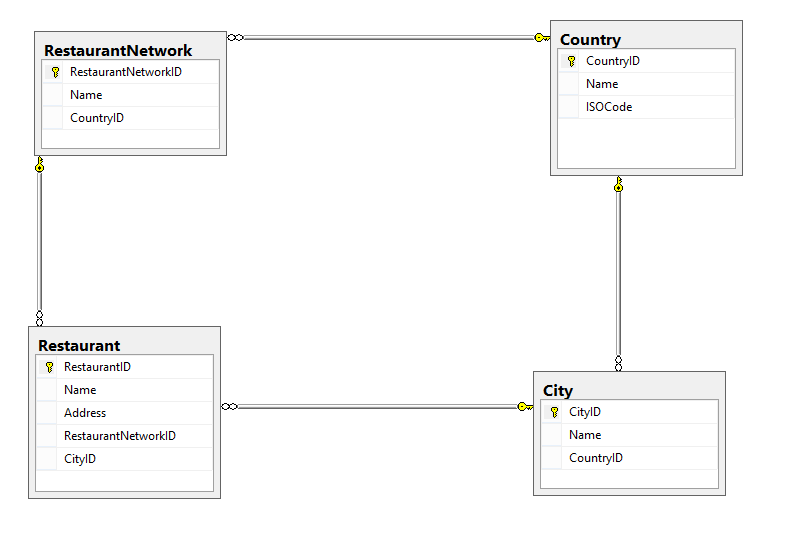


Рисунок 1. Модель бази данних

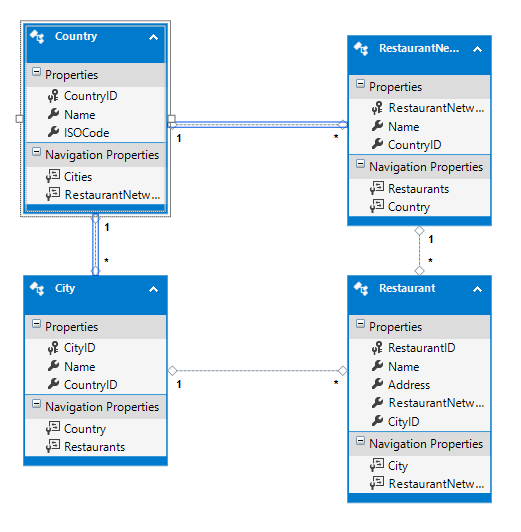


Рисунок 2. Entity Framework Модель

Лістінг компіляції і виконання

git init

msbuild Restoraunts.sln

Restoraunts\bin\debug\Restoraunts.exe add

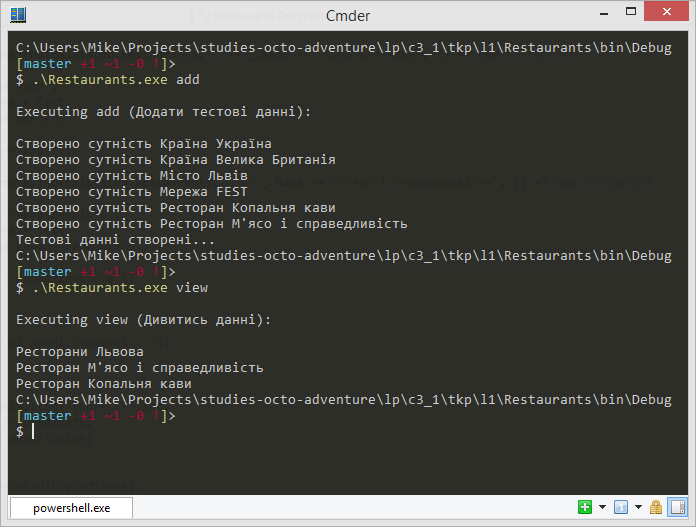
Restoraunts\bin\debug\Restoraunts.exe view

git add -A

git commit –m “Added implementation of the Restoraunts application”

Код на https://github.com/chaliy/studies-octo-adventure/tree/master/lp/c3\_1/trp/l1/

## Результат



## Висновки

Ознайомився зі структурованою мовою запитів SQL та з середовищем розробки баз даних Microsoft SQL Server 2008, здобув навики роботи в IDE Visual Studio 2010 з використанням об’єктно-реляційної проекції Entity Framework.