

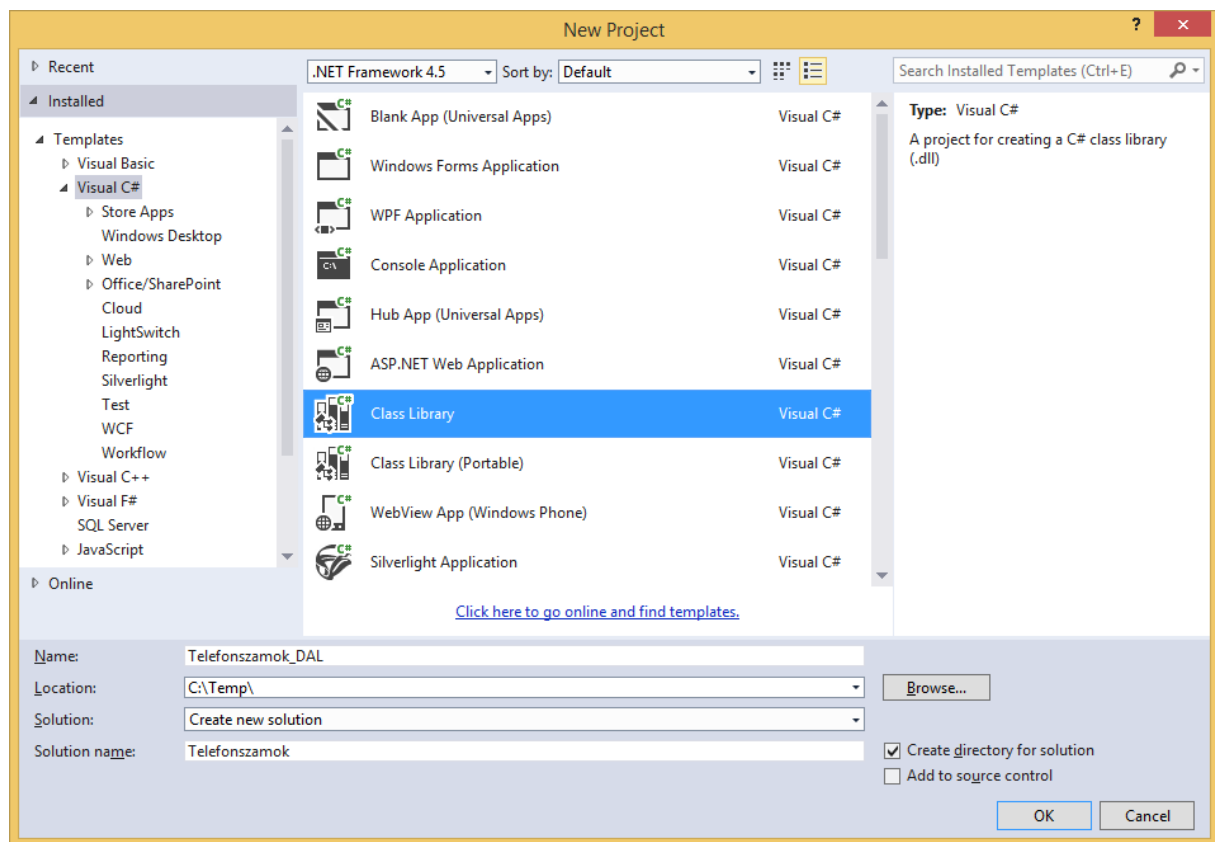
Entity Framework alapú adatbáziselérés

Dr. Johanyák Zsolt Csaba

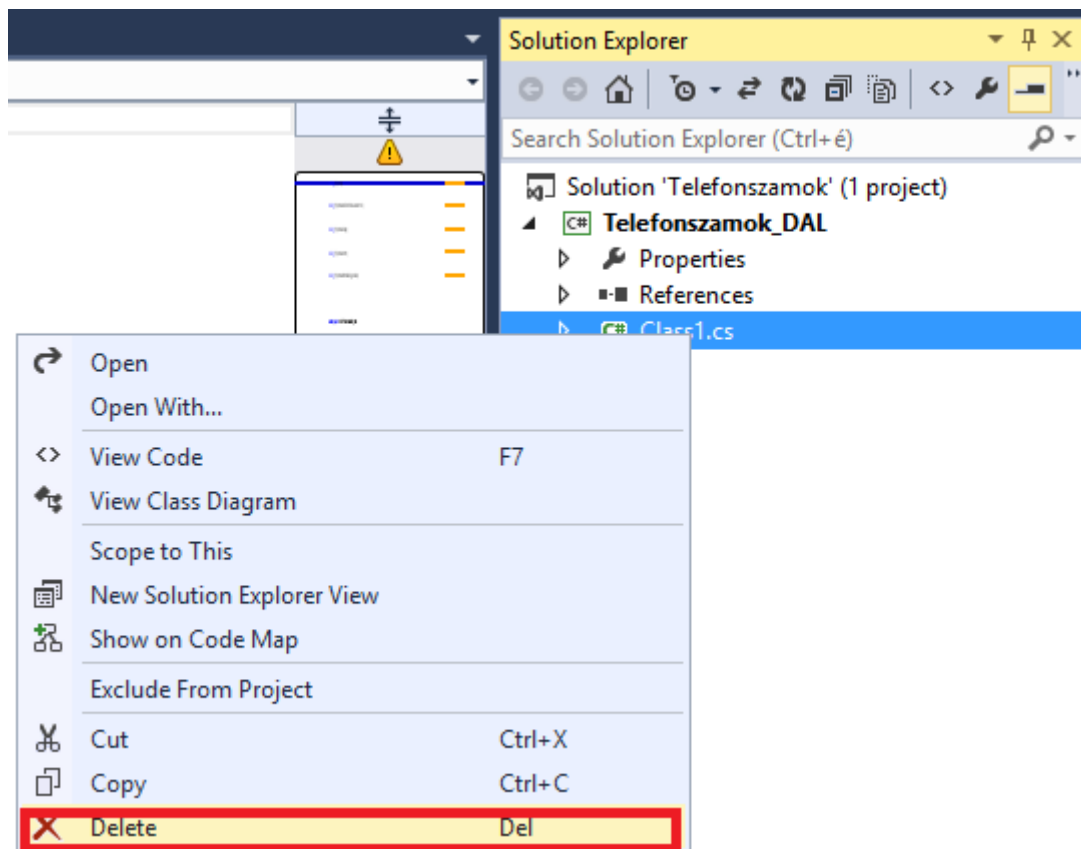
<http://johanyak.hu>

A gyakorlat célja Model-first megközelítéssel Entity-Framework modell létrehozása, majd ebből adatbázis generálása LocalDB-ben. adatok felvitele közvetlenül illetve programból, lekérdezések gyakorlása.

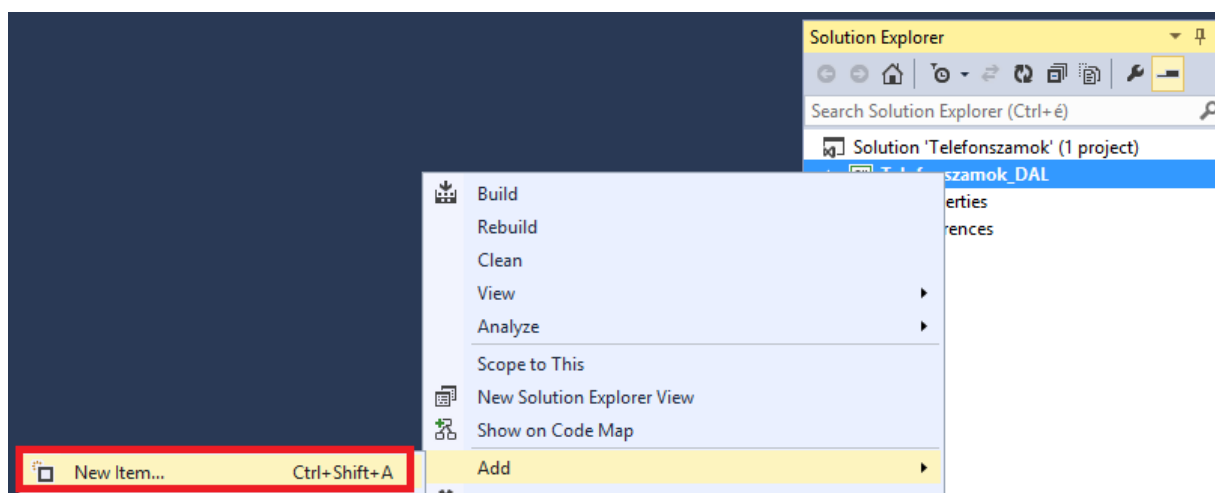
1. Az Entity Framework modell és az adatbázis létrehozása

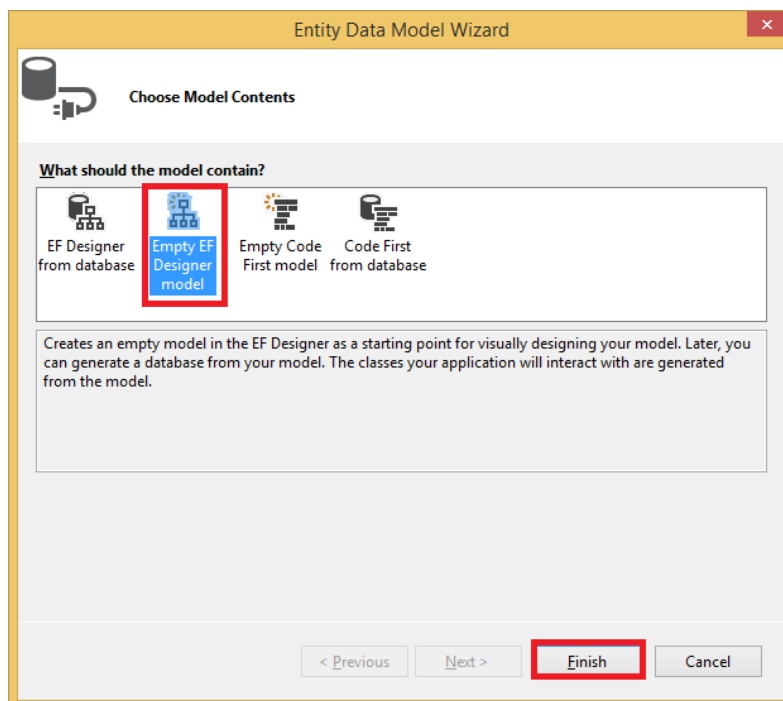
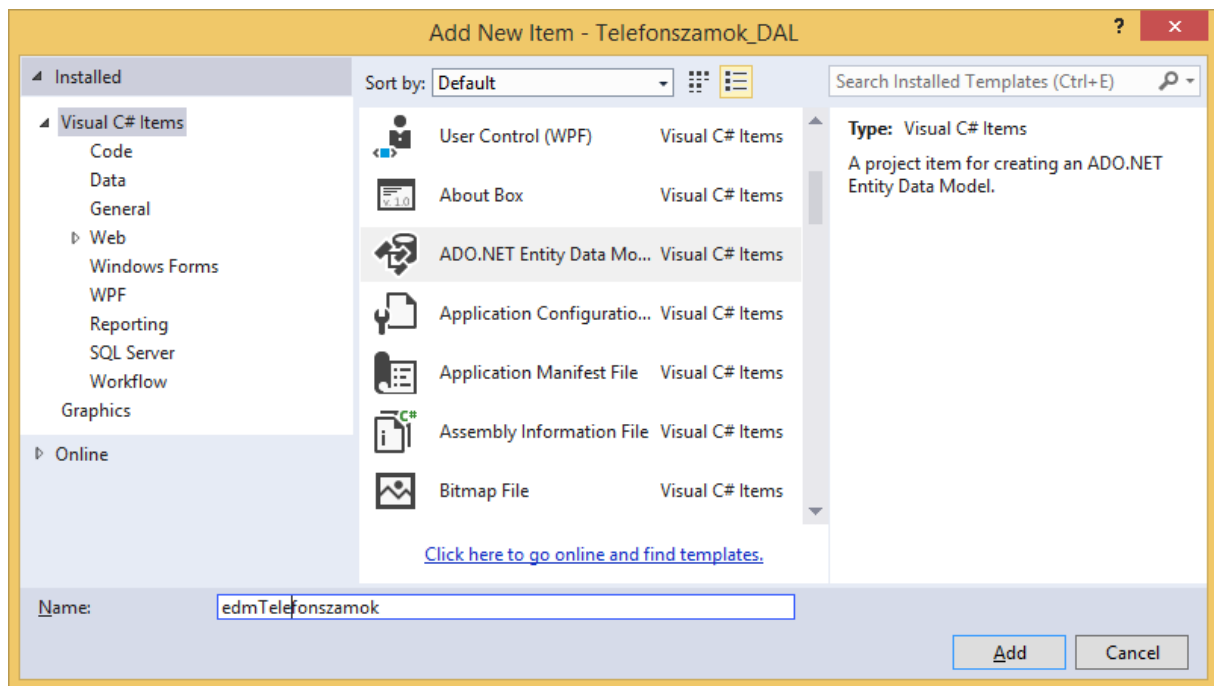


Töröljük a Class1.cs állományt.

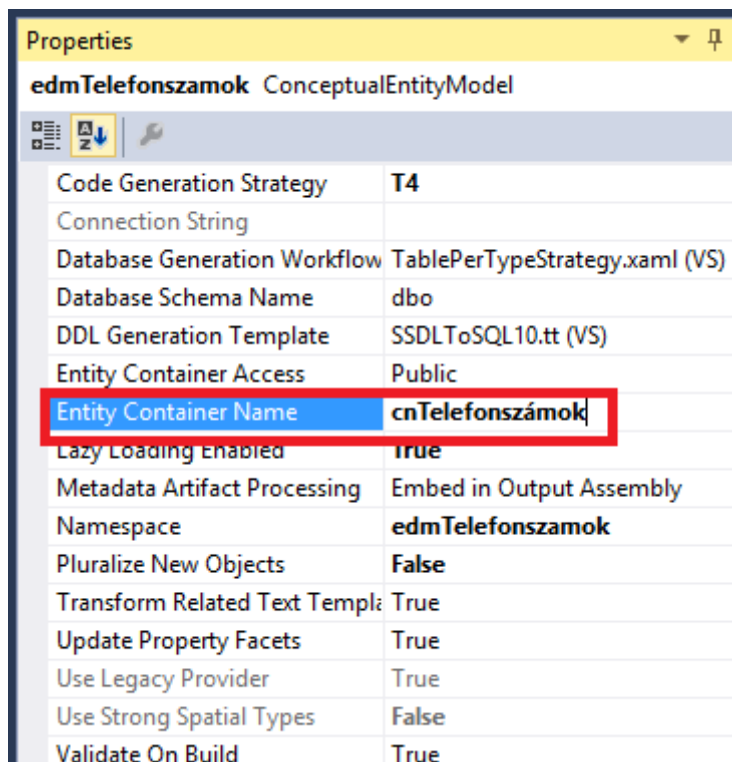


A projekthez adunk egy Entity Data Model-t.

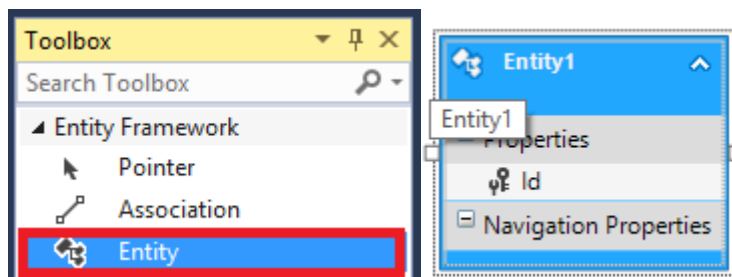




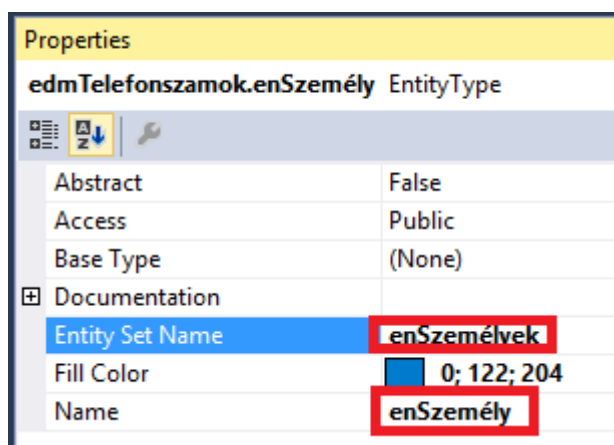
Megadjuk az entitás konténer osztály nevét.



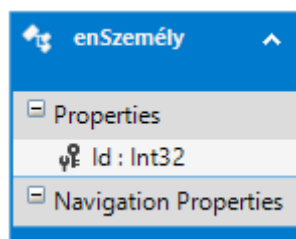
Létrehozunk egy enSzemély nevű entitást.



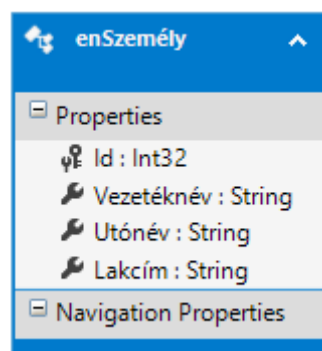
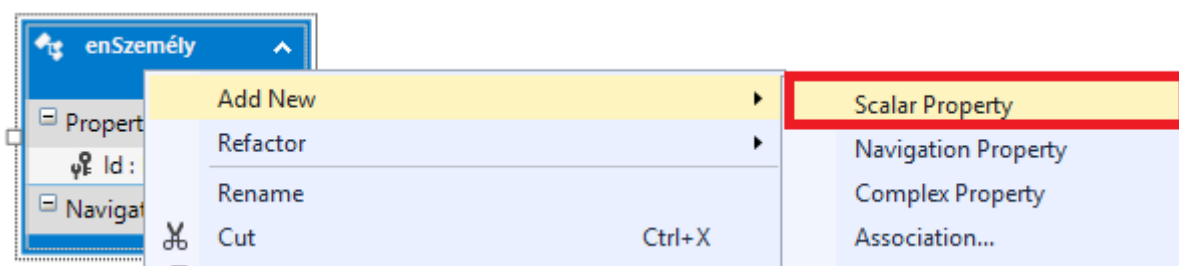
Beállítjuk a nevét és az entitáshalmaz nevét.



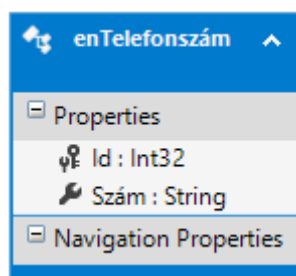
Beállítjuk, hogy a tulajdonságok nevei is megjelenjenek.



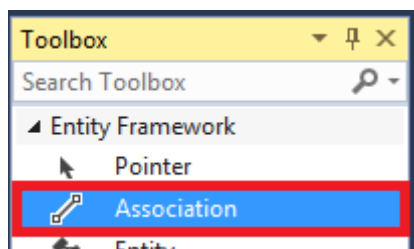
Hozzáadunk újabb tulajdonságokat.



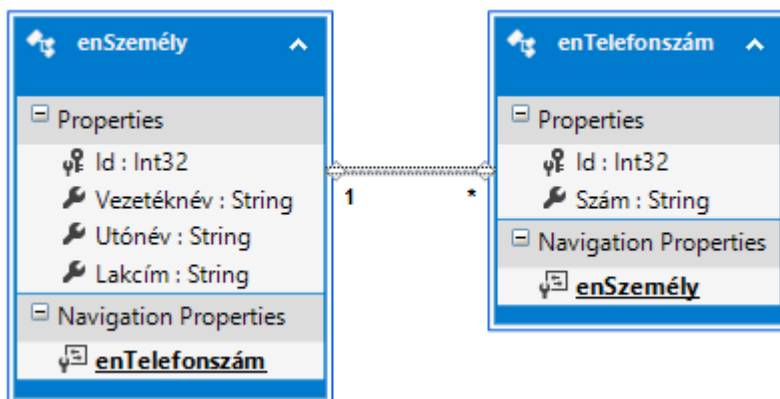
Egy személyhez több szám is tartozhat, ezért a számhoz létrehozunk egy külön entitást.



Összekötjük őket egy Association kapcsolattal az enSzemélytől kiindulva.



Navigációs tulajdonságok jelennek meg mindkét entitásnál.

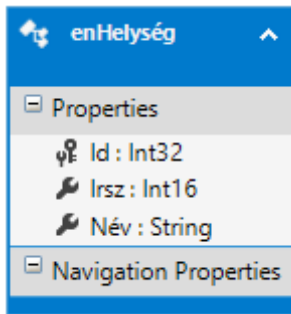


Az enSzemély oldalon a tulajdonság egy gyűjtemény, ezért a nevét többes számba tesszük a Properties ablakban.

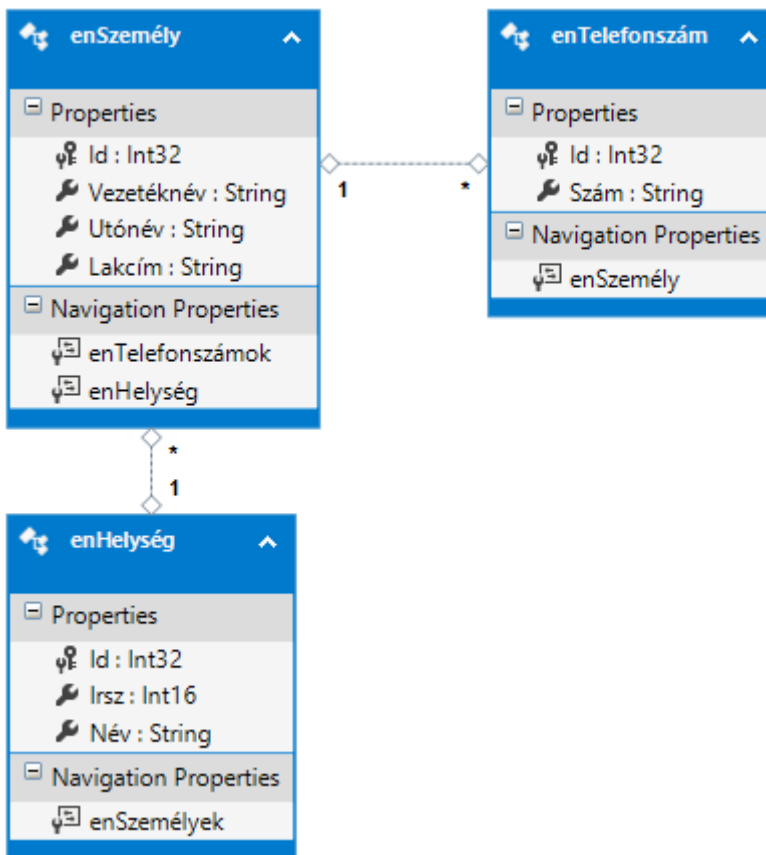
A screenshot of the Visual Studio Properties window. The 'Navigation Properties' section is expanded, showing the property 'edmTelefonszamok.enSzemély.enTelefonszám'. The 'Name' property is highlighted in blue, and its value 'enTelefonszamok' is highlighted in red.

Properties	
edmTelefonszamok.enSzemély.enTelefonszám NavigationProp...	
Association	enSzemélyenTelefonszám
Documentation	
From Role	enSzemély
Getter	Public
Multiplicity	* (Many)
Name	enTelefonszamok
Return Type	Collection of enTelefonszám
Setter	Public
To Role	enTelefonszám

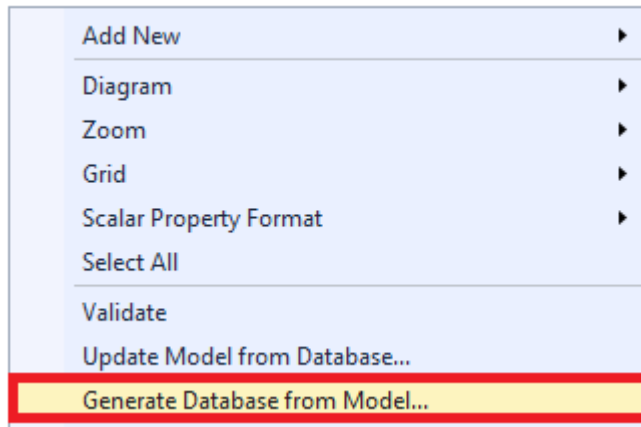
Létrehozunk egy entitást enHelység néven.




Egy helységben több személy is lakhat. Ennek megfelelően az enHelység felől indítjuk a kapcsolatot, majd itt is többes számba tesszük a személyeket.



Legeneráljuk az adatbázist.



Generate Database Wizard

 Choose Your Data Connection

Which data connection should your application use to connect to the database?

New Connection...

This connection string appears to contain sensitive data (for example, a password) that is required to connect to the database. Storing sensitive data in the connection string can be a security risk. Do you want to include this sensitive data in the connection string?

☐ No, exclude sensitive data from the connection string. I will set it in my application code.

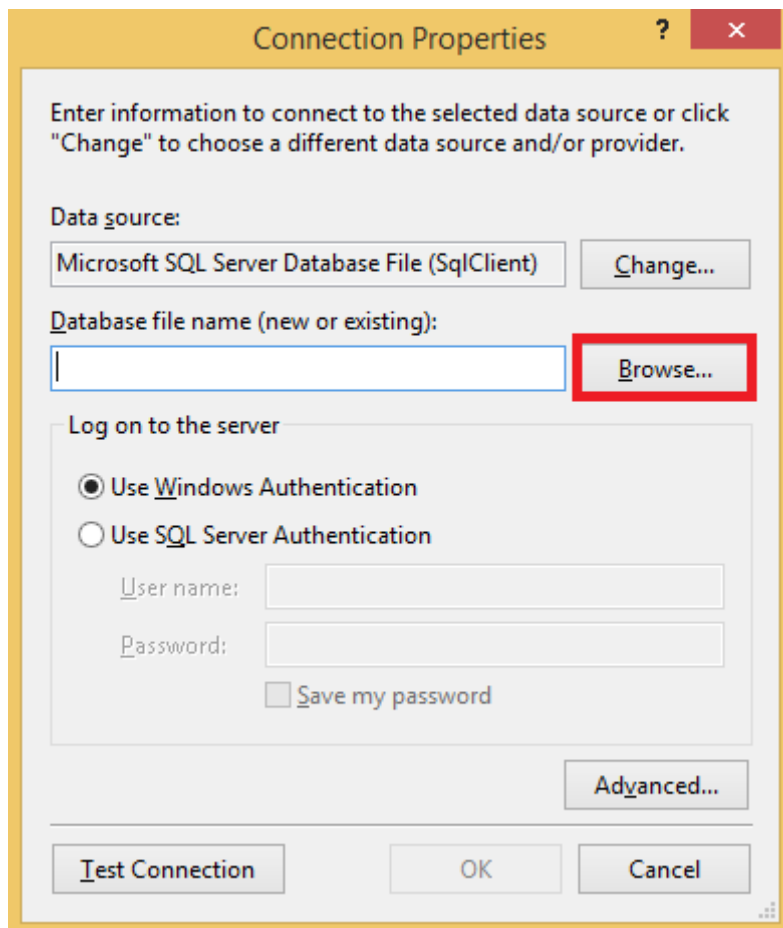
☐ Yes, include the sensitive data in the connection string.

Entity connection string:

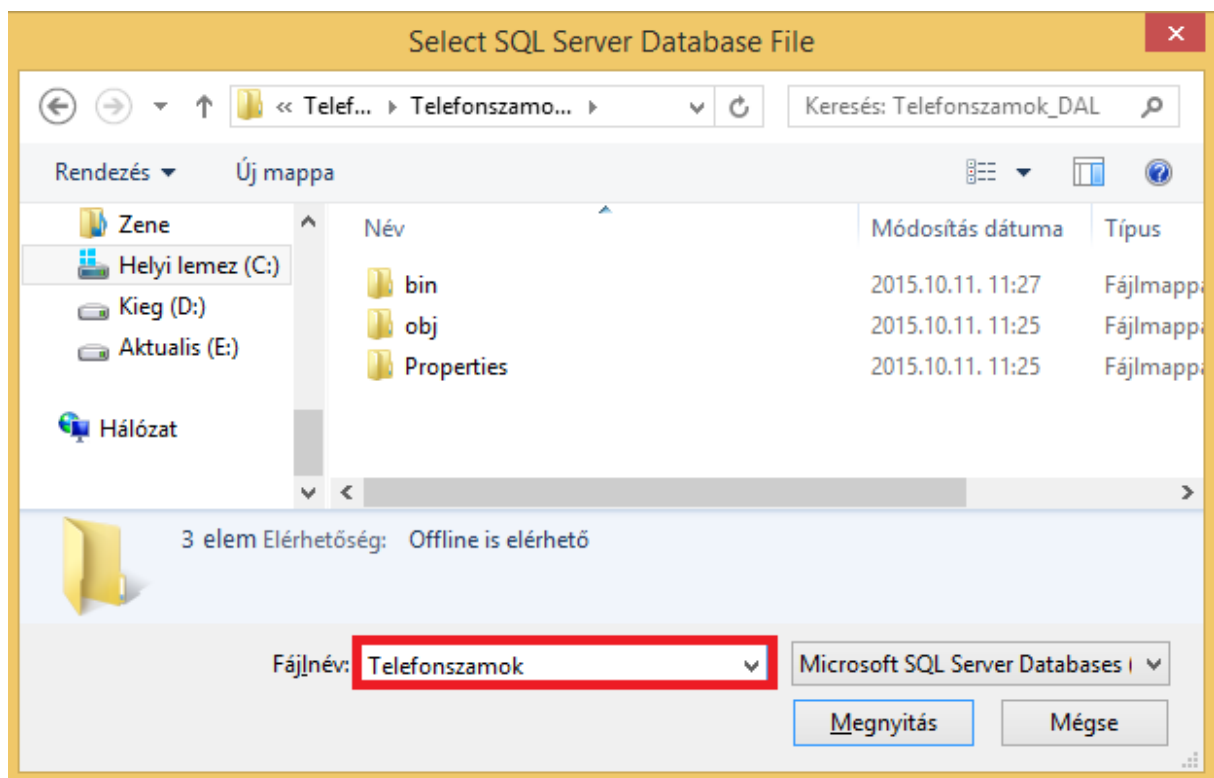
☒ Save entity connection settings in App.Config as:

< Previous Next > Finish Cancel

Az adatokat egy adatbázis állományban fogjuk tárolni a projekt könyvtárában.



Kiválasztjuk a projekt könyvtárát, majd megadjuk az állománynevet.



Connection Properties ? x

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source:
Microsoft SQL Server Database File (SqlClient) Change...

Database file name (new or existing):
C:\Temp\Telefonzamok\Telefonzamok_DAL\ Browse...

Log on to the server

☒ Use Windows Authentication

☐ Use SQL Server Authentication

User name:

Password:

☐ Save my password

Advanced...

Test Connection OK Cancel

Microsoft Visual Studio x

? The database file
'C:\Temp\Telefonzamok\Telefonzamok_DAL\Telefonzamok.mdf'
does not exist.

Would you like to create it?

Igen Nem

Generate Database Wizard



Choose Your Data Connection

Which data connection should your application use to connect to the database?

Telefonszamok.mdf



New Connection...

This connection string appears to contain sensitive data (for example, a password) that is required to connect to the database. Storing sensitive data in the connection string can be a security risk. Do you want to include this sensitive data in the connection string?

- ☐ No, exclude sensitive data from the connection string. I will set it in my application code.
- ☐ Yes, include the sensitive data in the connection string.

Connection string:

data source=(LocalDB)\v11.0;attachdbfilename=|DataDirectory|\Telefonszamok.mdf;integrated security=True;connect timeout=30;MultipleActiveResultSets=True;App=EntityFramework

☒ Save connection settings in App.Config as:

cnTelefonszamok

< Previous

Next >

Finish

Cancel

Generate Database Wizard



Choose Your Version

Which version of Entity Framework do you want to use?

☒ Entity Framework 6.x

☐ Entity Framework 5.0



It is also possible to install and use other versions of Entity Framework.

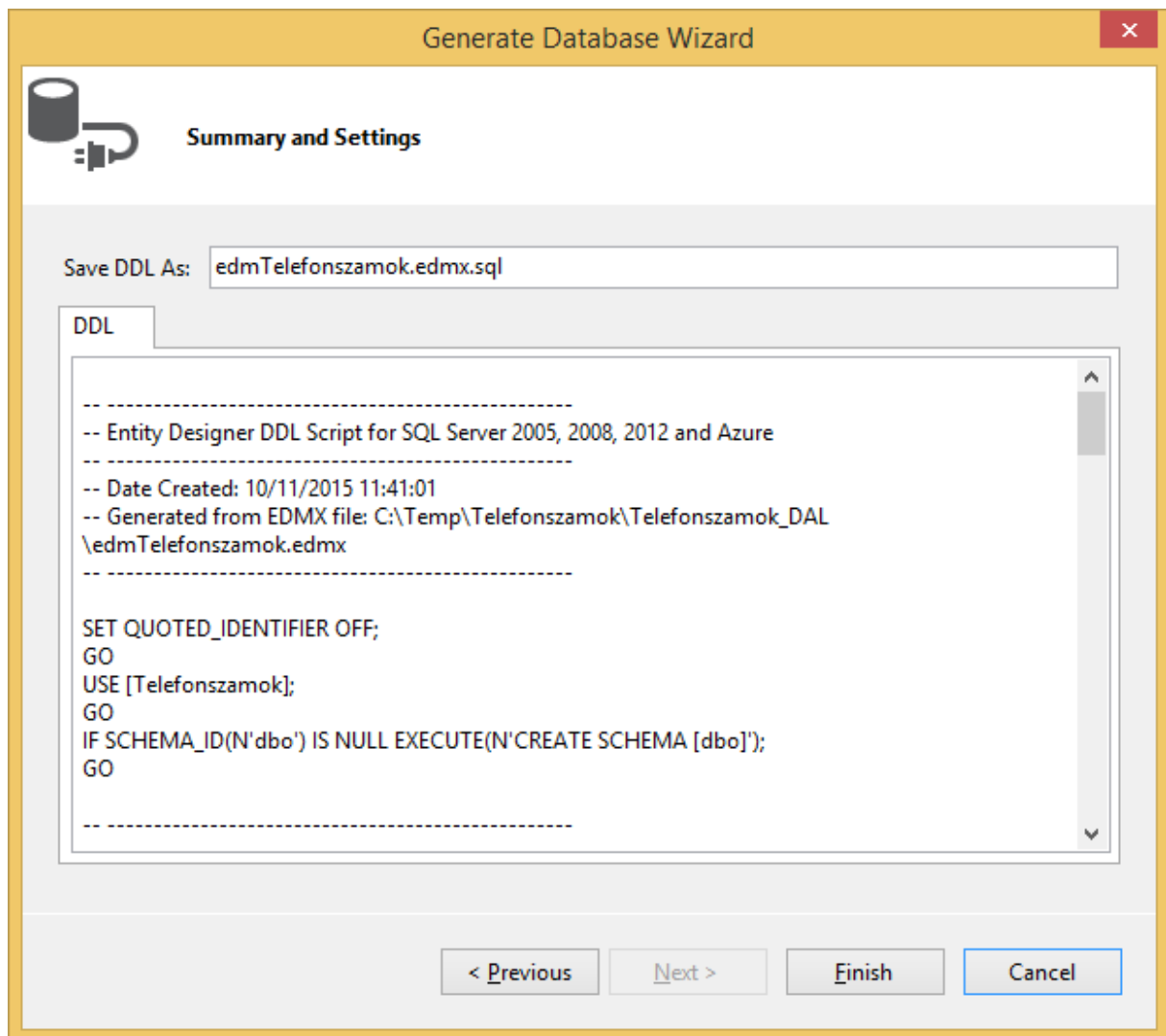
[Learn more about this](#)

< Previous

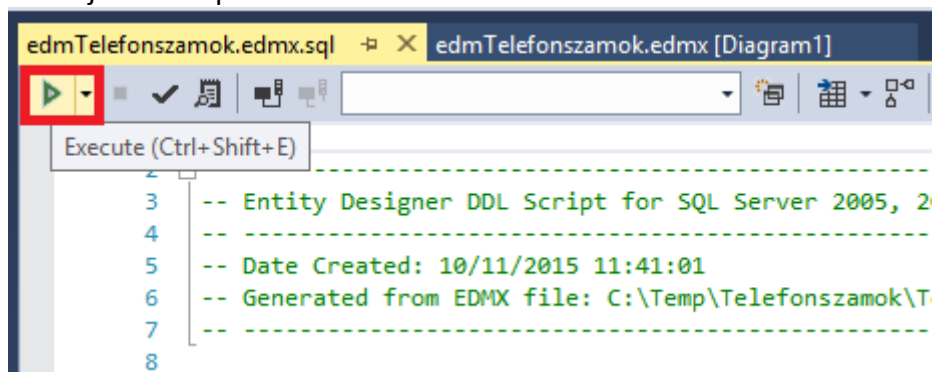
Next >

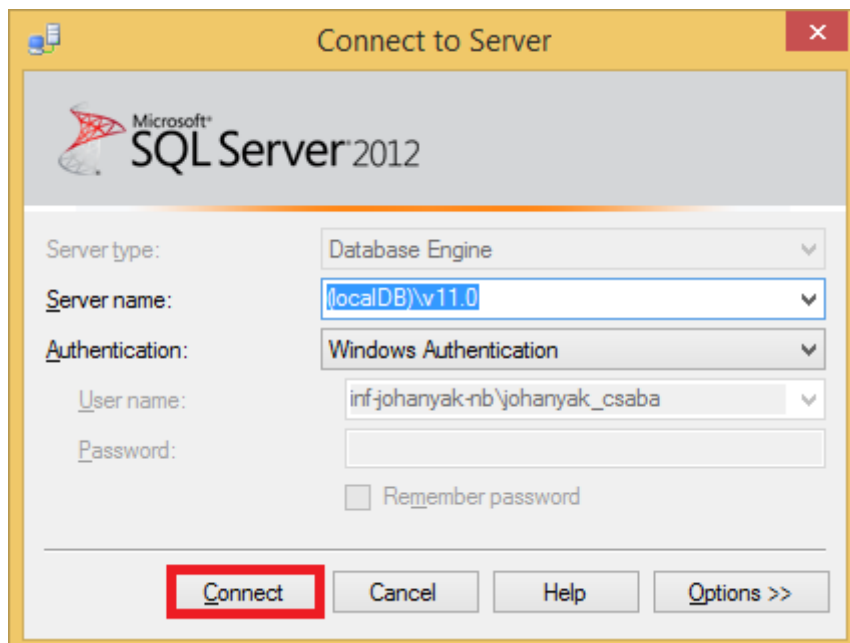
Finish

Cancel

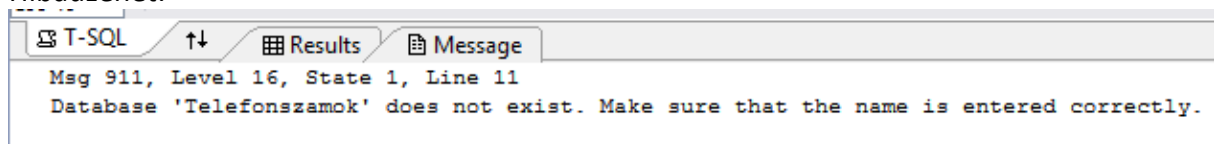


Futtatjuk a szkriptet.

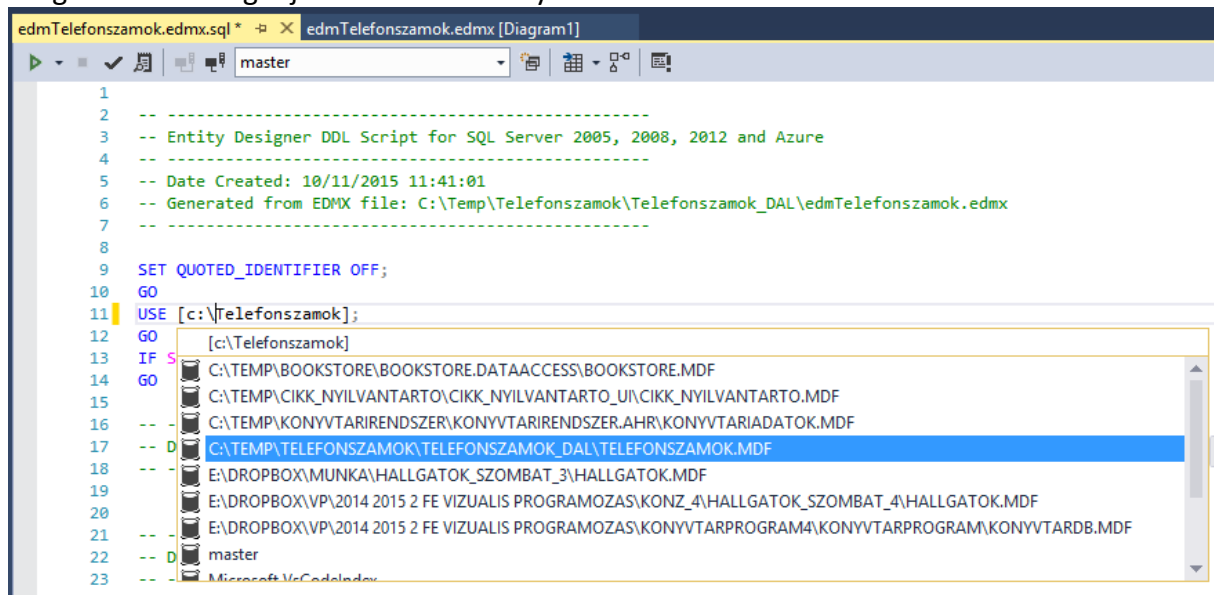




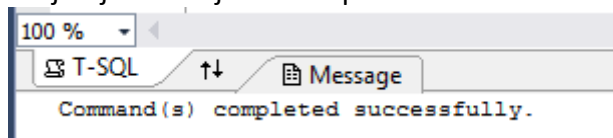
Hibaüzenet:



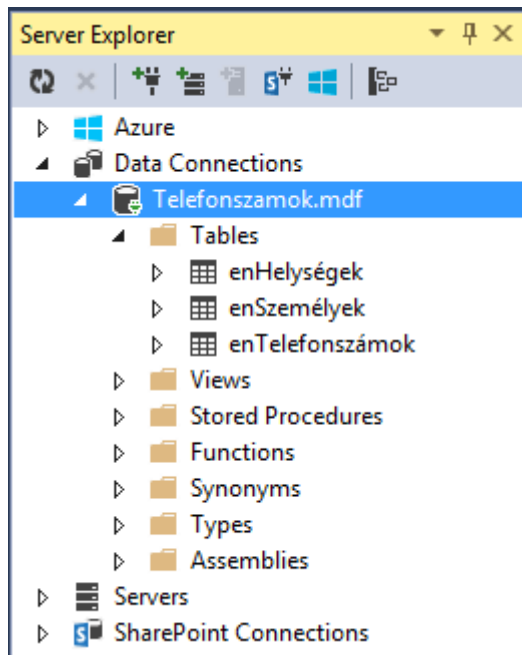
Megoldásként megadjuk az adatállomány elérési útvonalát.



Majd újból indítjuk a szkriptet. Ekkor már sikeres a végrehajtás.

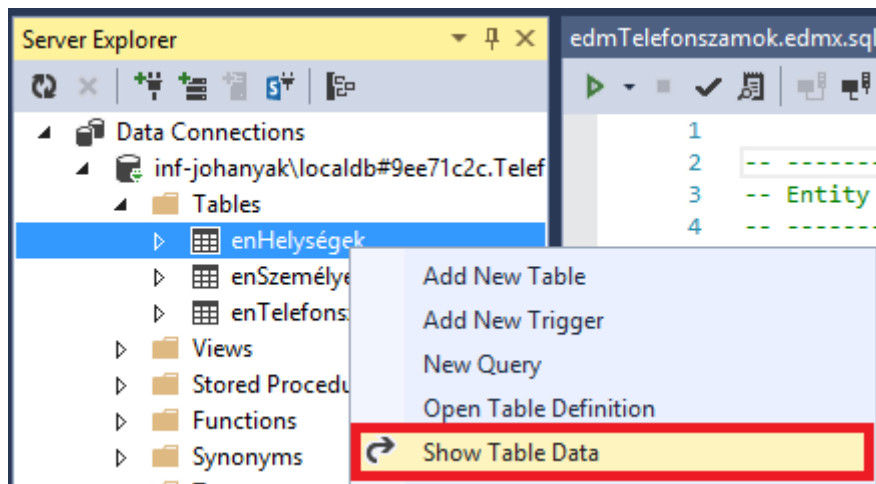


A Server Explorerben leellenőrizhetjük az adatbázis meglétét.



2. Adatfelvitel közvetlenül

Vigyünk fel adatokat az adatbázisba a Visual Studio segítségével.



Helységadatok. Csak a 2. és 3. oszlopba kell adatokat írni, az 1. oszlop automatikusan töltődik ki.

dbo.enHelységek [Data] X edmTelefonszamok.edmx.sql			
Max Rows: 1000			
	Id	Irsz	Név
	1	6000	Kecskemét
	2	2038	Sóskút
	3	2039	Pusztazámor
	4	2040	Budaörs
	5	2045	Törökbálint
▶*	NULL	NULL	NULL

Személyek

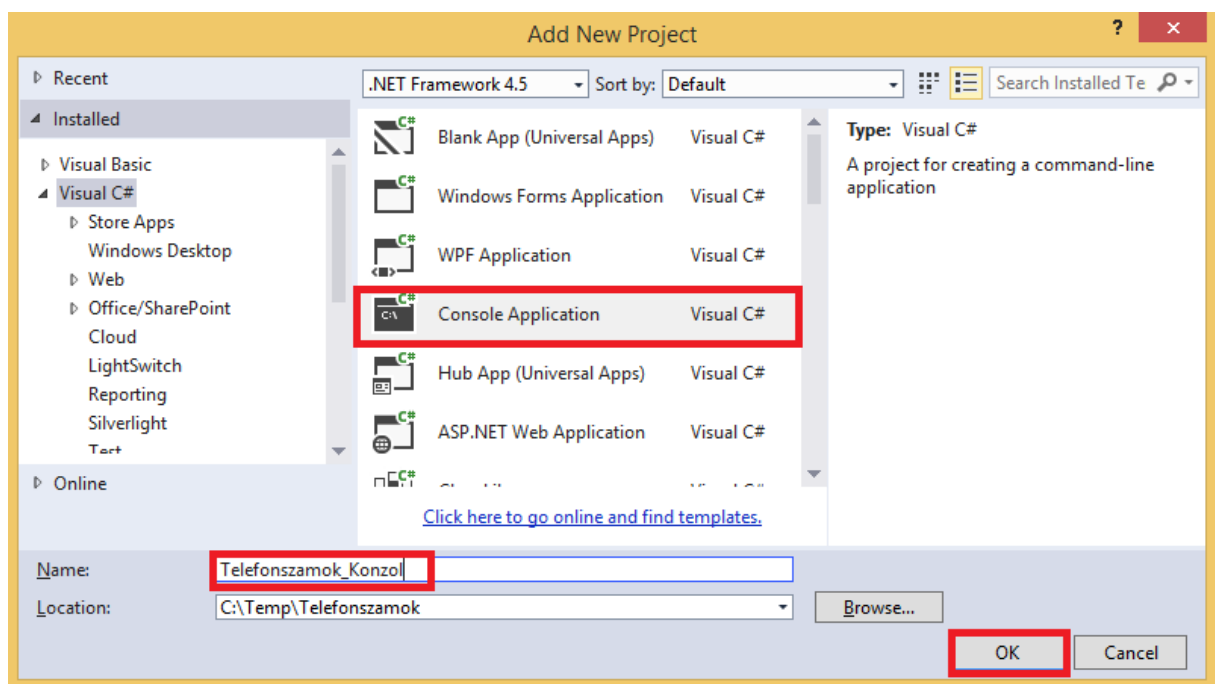
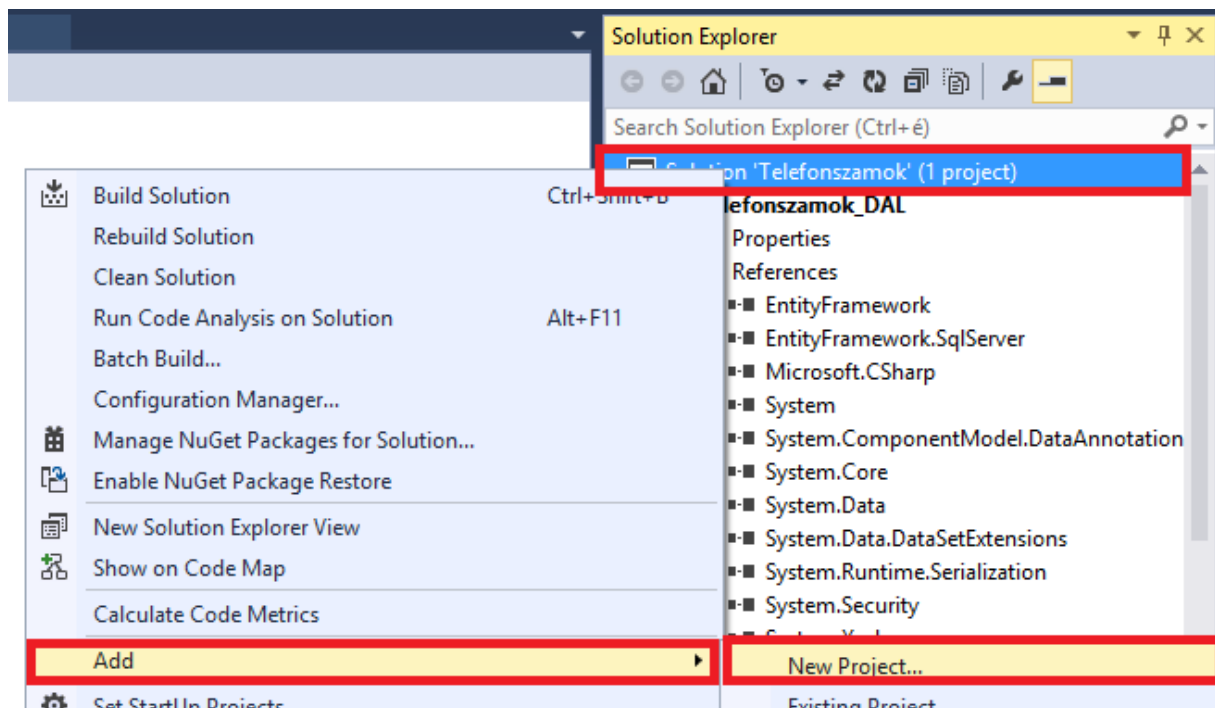
dbo.enTelefonszamok [Data] X dbo.enSzemélyek [Data] X dbo.enHelységek [Data]					
Max Rows: 1000					
	Id	Vezetéknév	Utónév	Lakcím	enHelység_Id
▶	1	Senki	Alfonz	Kis út 1.	1
	2	Gipsz	Jakab	Malom Köz 1.	1
	3	Erős	Áron	Alma rét 3.	2
	4	Olvasó	Jolán	Hermelin sugár...	3
★	NULL	NULL	NULL	NULL	NULL

Telefonszámok

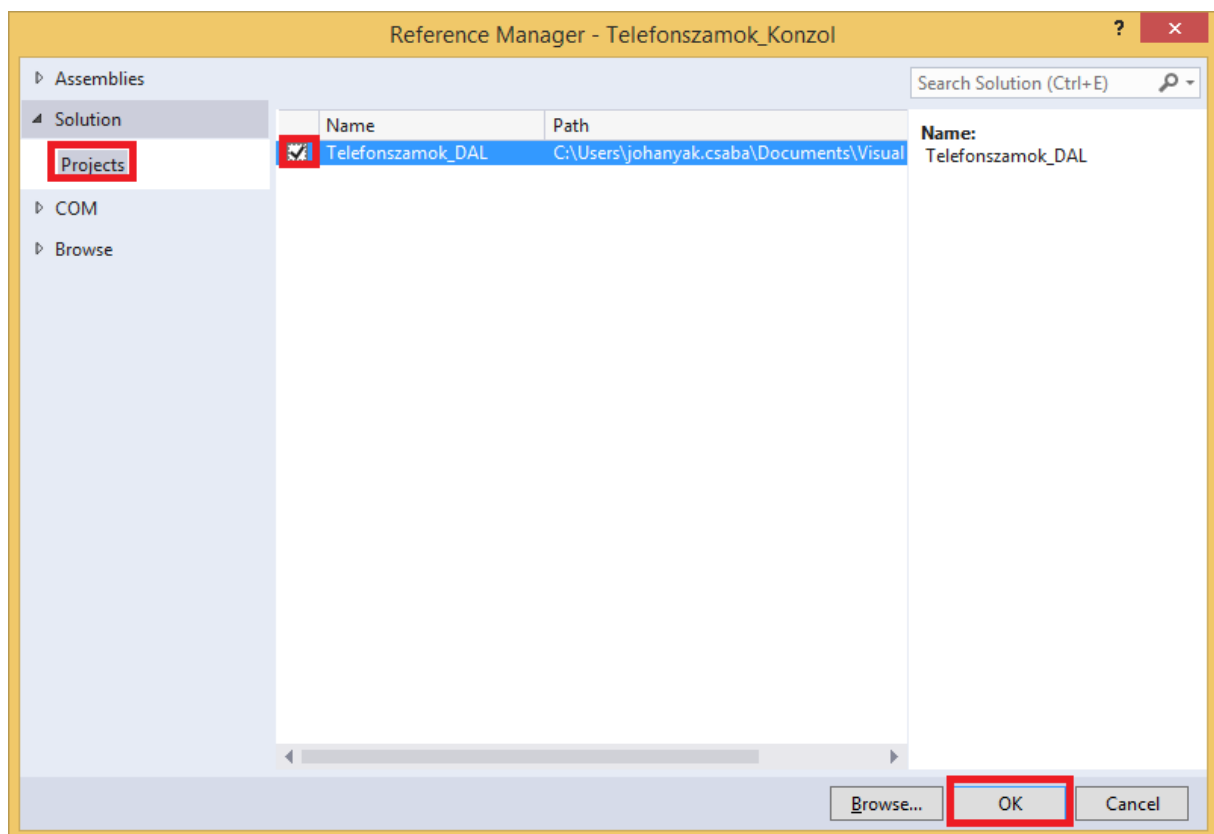
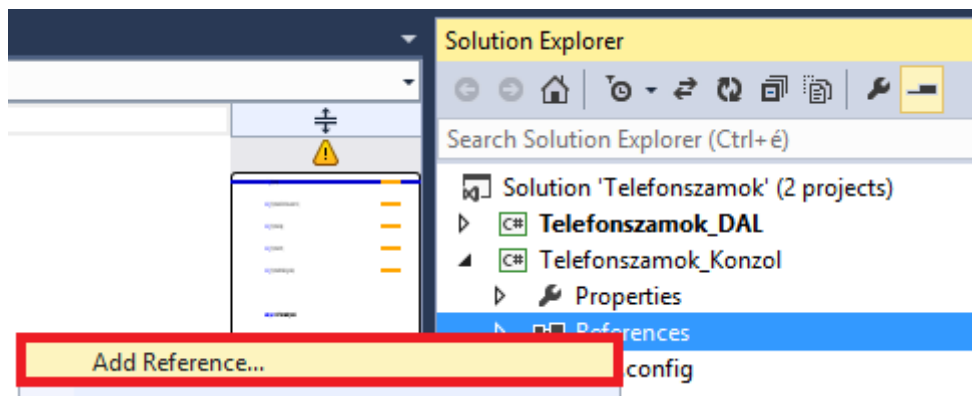
dbo.enTelefonszamok [Data] X dbo.enSzemélyek [Data]			
Max Rows: 1000			
	Id	Szám	enSzemély_Id
	1	+36-11-555-5555	1
	2	+36-12-444-4444	1
	3	+36-11-333-3333	4
	5	+36-11-111-1111	3
▶*	NULL	NULL	NULL

3. Adatfelvitel programból

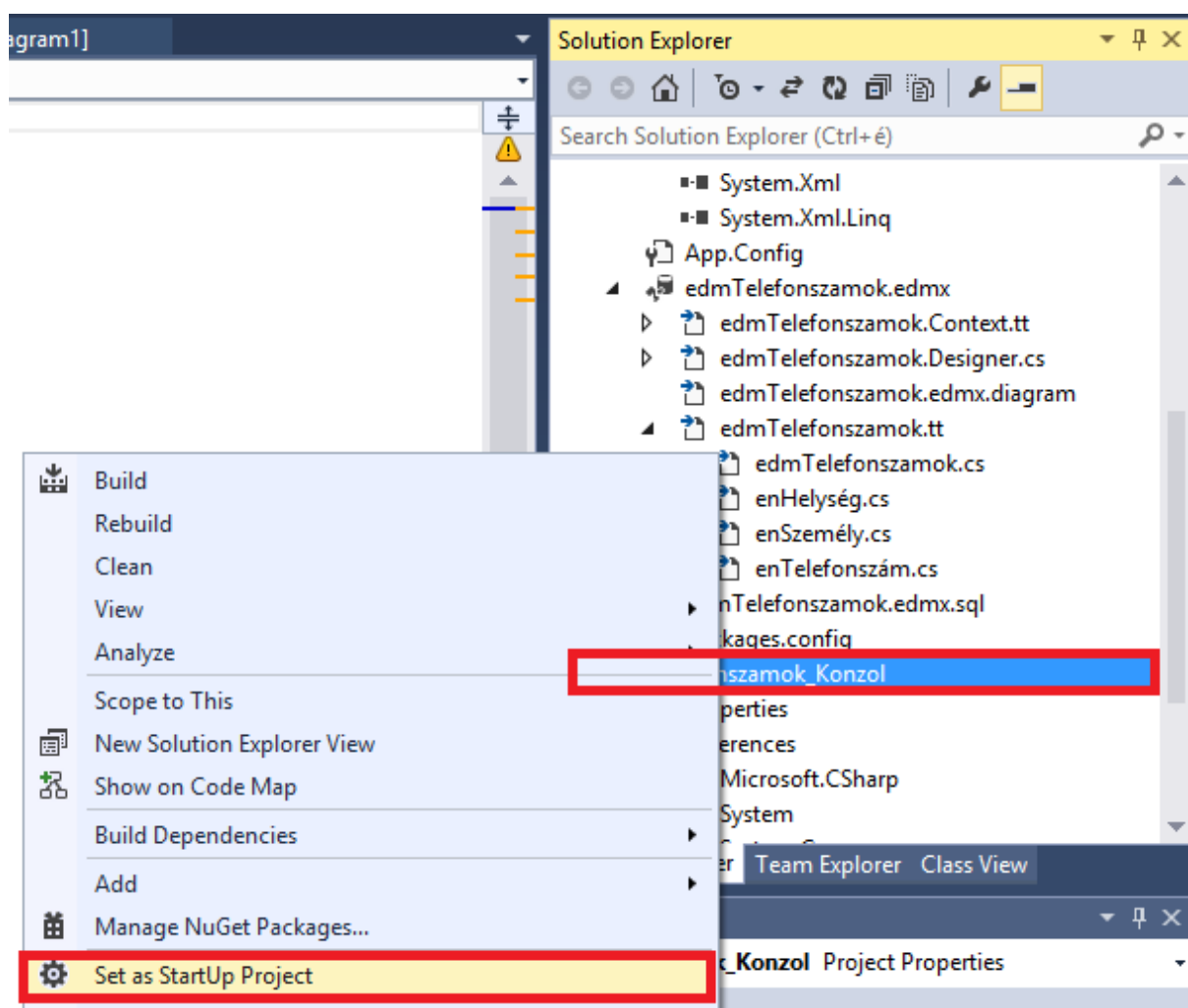
Hozzunk létre egy konzolalkalmazás projektet a megoldáson belül.



Vegyük fel a referenciák közé az osztálykönyvtár projektünket.



Állítsuk be indító projektnek a konzol projektet.



Állítsuk be a DAL projekt névtérét felhasznált névtérként a konzol alkalmazásban.

```

Program.cs*  X  edmTelefonsszamok.edmx [Diagram1]*
Telefonsszamok_Konzol.Program

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6  using Telefonsszamok DAL;
7
8  namespace Telefonsszamok_Konzol

```

Létrehozunk egy adattagot az entitáskonténer számára, majd felvesszük a referenciák közé az Entity Framework-öt. Tegyük az adattagot statikussá.

```

8 namespace Telefonszamok_Konzol
9 {
10
11     class Program
12     {
13         static cnTelefonszámok cnTelefonszámok;
14         static void Main(string[] args)
15         {
16

```

Hozzunk létre egy konténer objektumot. Készítsünk egy statikus metódust adatfelvitel céljára.

```

14 static void Main(string[] args)
15 {
16     cnTelefonszámok = new cnTelefonszámok();
17     Adatfelvitel();
18 }
19

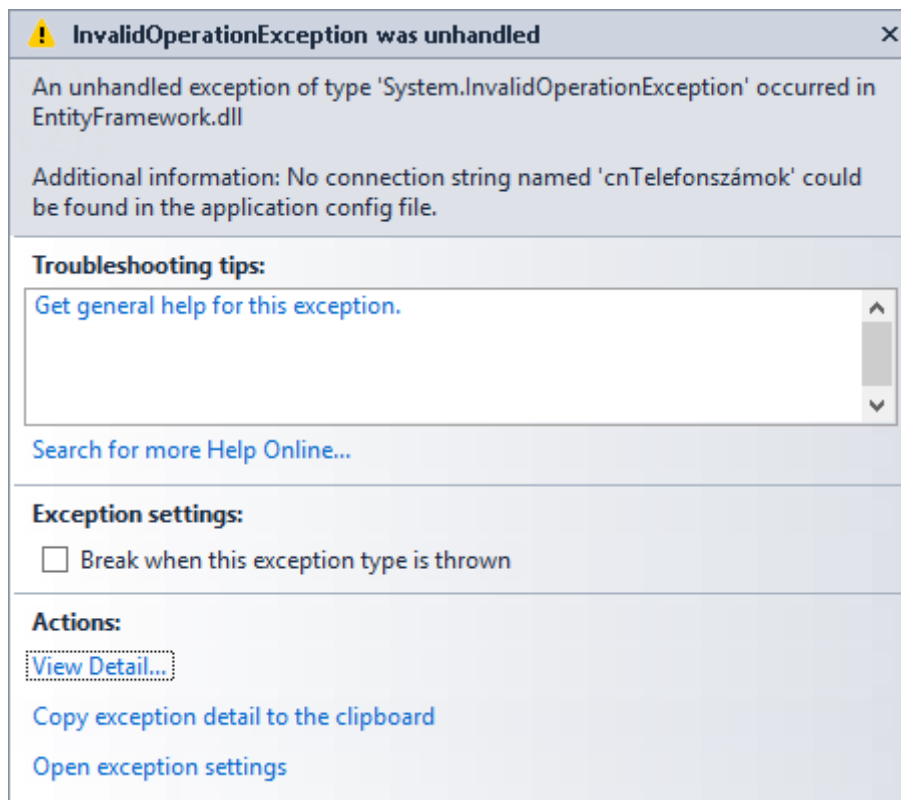
```

```

private static void Adatfelvitel()
{
    var h = new enHelység {Irsz = 2090, Név = "Remeteszőlős"};
    var sz = new enSzemély
    {
        Vezetéknév = "Argon",
        Utónév = "Géza",
        enHelység = h,
        Lakcím = "Ordas köz 6."
    };
    h.enSzemélyek.Add(sz);
    var t1 = new enTelefonszám {Szám = "+36-13-555-555", enSzemély = sz };
    var t2 = new enTelefonszám {Szám = "+36-13-556-555", enSzemély = sz };
    sz.enTelefonszámok.Add(t1);
    sz.enTelefonszámok.Add(t2);
    cnTelefonszámok.enHelységek.Add(h);
    cnTelefonszámok.enSzemélyek.Add(sz);
    cnTelefonszámok.enTelefonszámok.Add(t1);
    cnTelefonszámok.enTelefonszámok.Add(t2);
    cnTelefonszámok.SaveChanges();
}
}

```

Futtassuk le a programot, majd ellenőrizzük le, hogy az adatbázisba kerültek-e az adatok.



Hiba: nincs kapcsolatunk az adatbázishoz. Megoldás: másoljuk át az App.Config állományt az osztálykönyvtár projektből a konzolalkalmazás projektjébe, majd a konzolalkalmazás Main metódusának elejére helyezzük el az alábbi két utasítást.

```
var DataDir =  
    Directory.GetParent(Directory.GetParent(Directory.GetParent(  
        Directory.GetCurrentDirectory()).FullName).FullName) +  
    "\\Telefonszamok_DAL";
```

```
AppDomain.CurrentDomain.SetData("DataDirectory", DataDir);
```

Indítsuk el a konzolalkalmazást. Amennyiben hibaüzenet nélkül lefut, akkor tegyük megjegyzésbe az Adatfelvitel(); utasítást, mert erre már nem lesz szükségünk a továbbiakban, az adatok bekerültek az adatbázisba.

Ellenőrizzük le a Server Explorer segítségével, hogy sikerült-e az adatfelvitel.

	Id	Irsz	Név
▶	1	6000	Kecskemét
	2	2038	Sóskút
	3	2039	Pusztazámor
	4	2040	Budaörs
	5	2045	Törökbálint
	6	2090	Remeteszőlős
★	NULL	NULL	NULL

Server Explorer					
<div> <div>Azure</div> <div>Data Connections</div> <div>Telefonszamok.mdf</div> <div>Tables</div> <div>enHelységek</div> <div>enSzemélyek</div> <div>enTelefonszámok</div> <div>Views</div> <div>Stored Procedures</div> </div>					
dbo.enTelefonszámok [Data]					
Max Rows: 1000					
	Id	Vezetéknév	Utónév	Lakcím	enHelység_Id
	1	Senki	Alfonz	Kis út 1.	1
	2	Gipsz	Jakab	Malom Köz 1.	1
	3	Erős	Áron	Alma rét 3.	2
	4	Olvasó	Jolán	Hermelin sugár...	3
	5	Argon	Géza	Ordas köz 6.	6
*	NULL	NULL	NULL	NULL	NULL

Server Explorer			
<div> <div>Azure</div> <div>Data Connections</div> <div>Telefonszamok.mdf</div> <div>Tables</div> <div>enHelységek</div> <div>enSzemélyek</div> <div>enTelefonszámok</div> <div>Views</div> <div>Stored Procedures</div> <div>Functions</div> </div>			
dbo.enTelefonszámok [Data]			
Max Rows: 1000			
	Id	Szám	enSzemély_Id
	1	+36-11-555-5555	1
	2	+36-12-444-4444	1
	3	+36-11-333-3333	4
	5	+36-11-111-1111	3
	6	+36-13-555-555	5
	7	+36-13-556-555	5
▶*	NULL	NULL	NULL

4. Lekérdezések programból

Készítsünk egy lekérdező metódust, ami a lekérdezés eredményét kiírja a konzolra. A metódusban vegyük sorra a személyeket, és írassuk ki az adataikat. A telefonszámok egymástól vesszővel elválasztva jelenjenek meg. Hívjuk meg a Lekérdez() metódust a Main() metódusból.

```
private static void Lekérdez()
{
    Console.WriteLine("Összes adat\r\n-----");
    foreach (var x in cnTelefonszámok.enSzemélyek)
    {
        var s = x.Vezetéknév + " " + x.Utónév + ": " + x.enHelység.Irsz +
            " " + x.enHelység.Név + ", " + x.Lakcím + ", ";
        foreach (var y in x.enTelefonszámok)
        {
            s += y.Szám;
            if (y != x.enTelefonszámok.Last())
                s += ", ";
        }
        Console.WriteLine(s);
    }
}
```

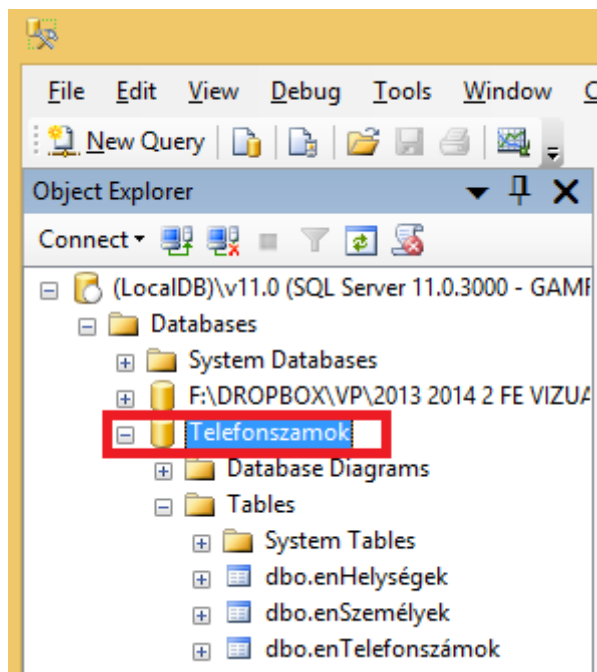
Futtassuk le a programot.

```
file:///C:/Temp/Telefonszamok/Telefonszamok_Konzol/bin/Debug/Telefonsza...
Összes adat
-----
Senki Alfonz: 6000 Kecskemét, Kis út 1., +36-11-555-5555, +36-12-444-4444
Gipsz Jakab: 6000 Kecskemét, Malom Köz 1.,
Erős Áron: 2038 Sósút, Alma rét 3., +36-11-111-1111
Olvasó Jolán: 2039 Pustazámor, Hermelin sugárút 5., +36-11-333-3333
Argon Géza: 2090 Remeteszőlős, Ordas köz 6., +36-13-555-555, +36-13-556-555
```

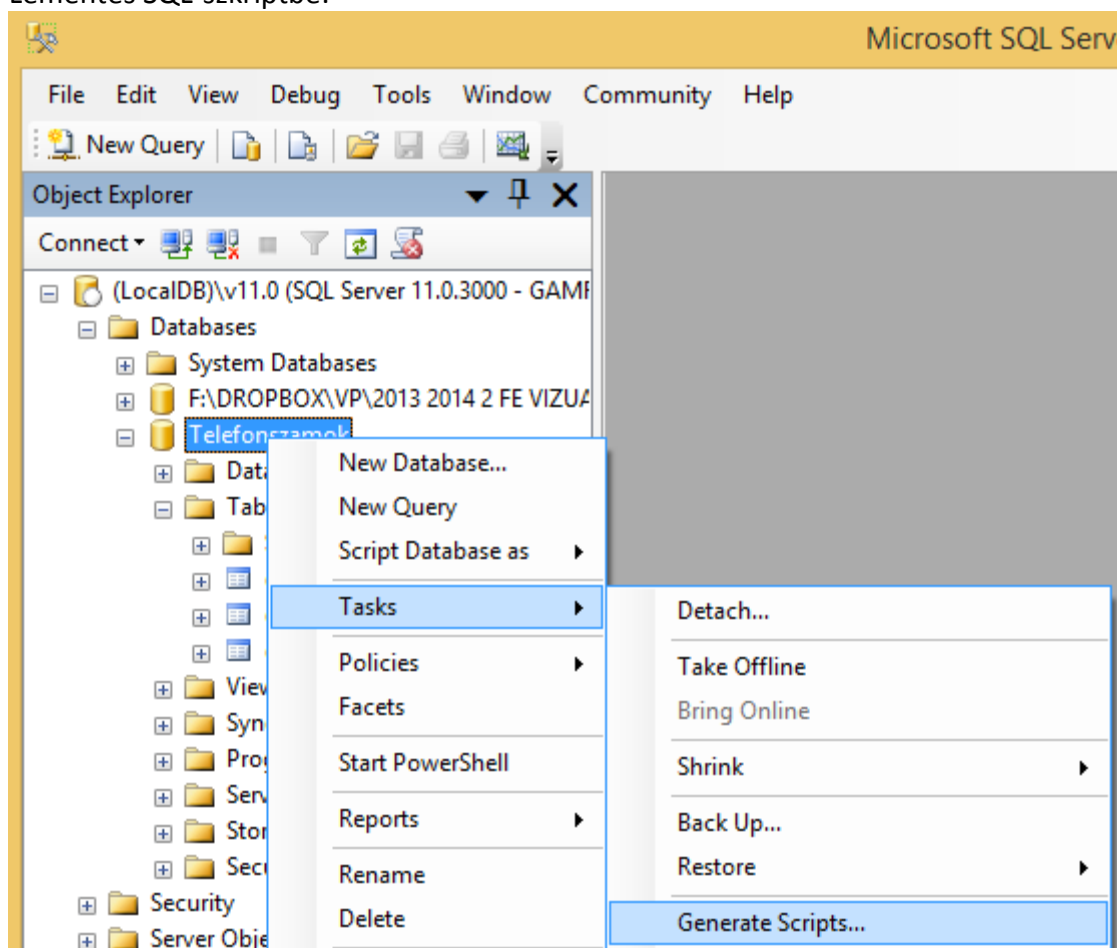
5. Az adatbázisban tárolt adatok lementése egy SQL szkriptbe az adatbázis szerkezettel együtt, majd az adatbázis újbóli létrehozása.


A gyakorlat célja az, hogy áttekinthessük, hogy hogyan tudjuk az adatbázist átvinni egy másik gépre/kiszolgálóra az adatbázis állomány (Telefonszamok.mdf) mozgatása/másolása nélkül. A feladat végrehajtásához a Microsoft SQL Server Management Studio-t és a Visual Studio-t használjuk. Indítsuk el a Microsoft SQL Server Management Studio-t.





Lementés SQL-szkriptbe:





Introduction

Introduction

Choose Objects

Set Scripting Options

Summary

Save or Publish Scripts


Help

Generate scripts for database objects.

This wizard generates a script of selected database objects. The scripts can be saved for later use in creating databases in an instance of the Database Engine, or to publish a database to a Database Publishing Web Service.

There are four steps to complete this wizard:

1. Select database objects.
2. Specify scripting or publishing options.
3. Review your selections.
4. Generate scripts, then save or publish them.



To begin the script generation process, click Next.

☐ Do not show this page again.


< Previous

Next >

Finish

Cancel

Generate and Publish Scripts



Choose Objects

Introduction

Choose Objects

Set Scripting Options

Summary

Save or Publish Scripts

Help

Select the database objects to script.

☒ Script entire database and all database objects

☐ Select specific database objects

Tables

Select All

Deselect All


< Previous

Next >

Finish

Cancel

Generate and Publish Scripts



Set Scripting Options

Introduction

Choose Objects

Set Scripting Options

Summary

Save or Publish Scripts

Help

Specify how scripts should be saved or published.

Output Type

☒ Save scripts to a specific location

☐ Publish to Web service

☒ Save to file

Advanced

Files to generate:

☒ Single file

☐ Single file per object

File name:

F:\Dropbox\WP\2013 2014 2 fe Vizualis program

...

☒ Overwrite existing file

Save as:

☒ Unicode text

☐ ANSI text

☐ Save to Clipboard

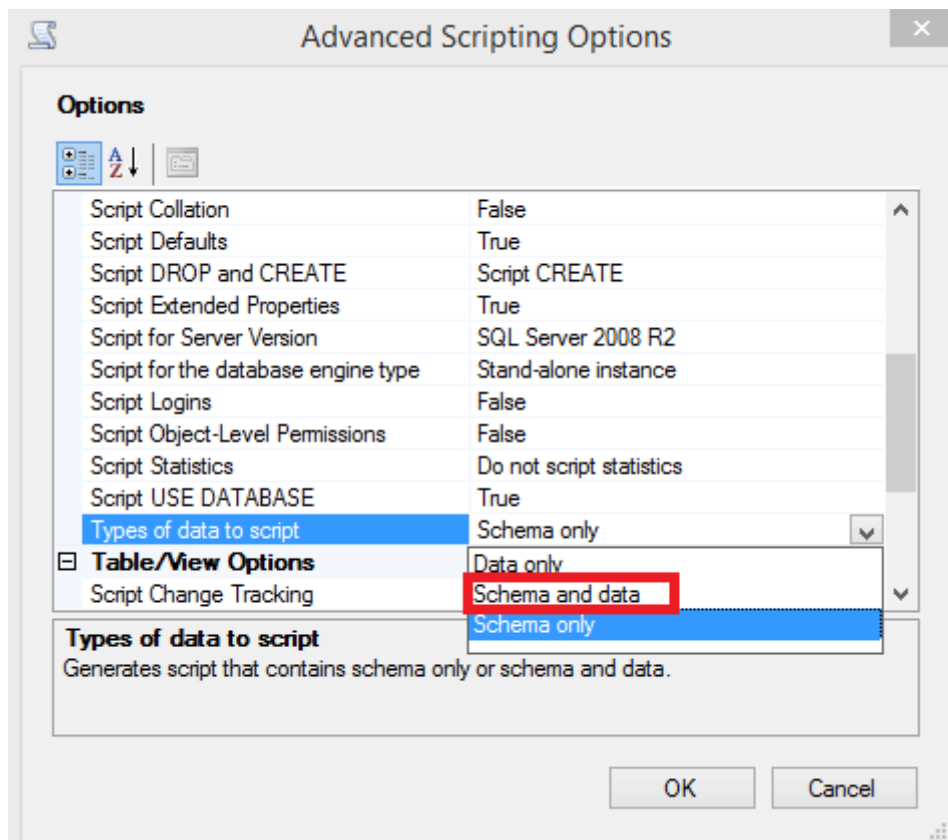
☐ Save to new query window

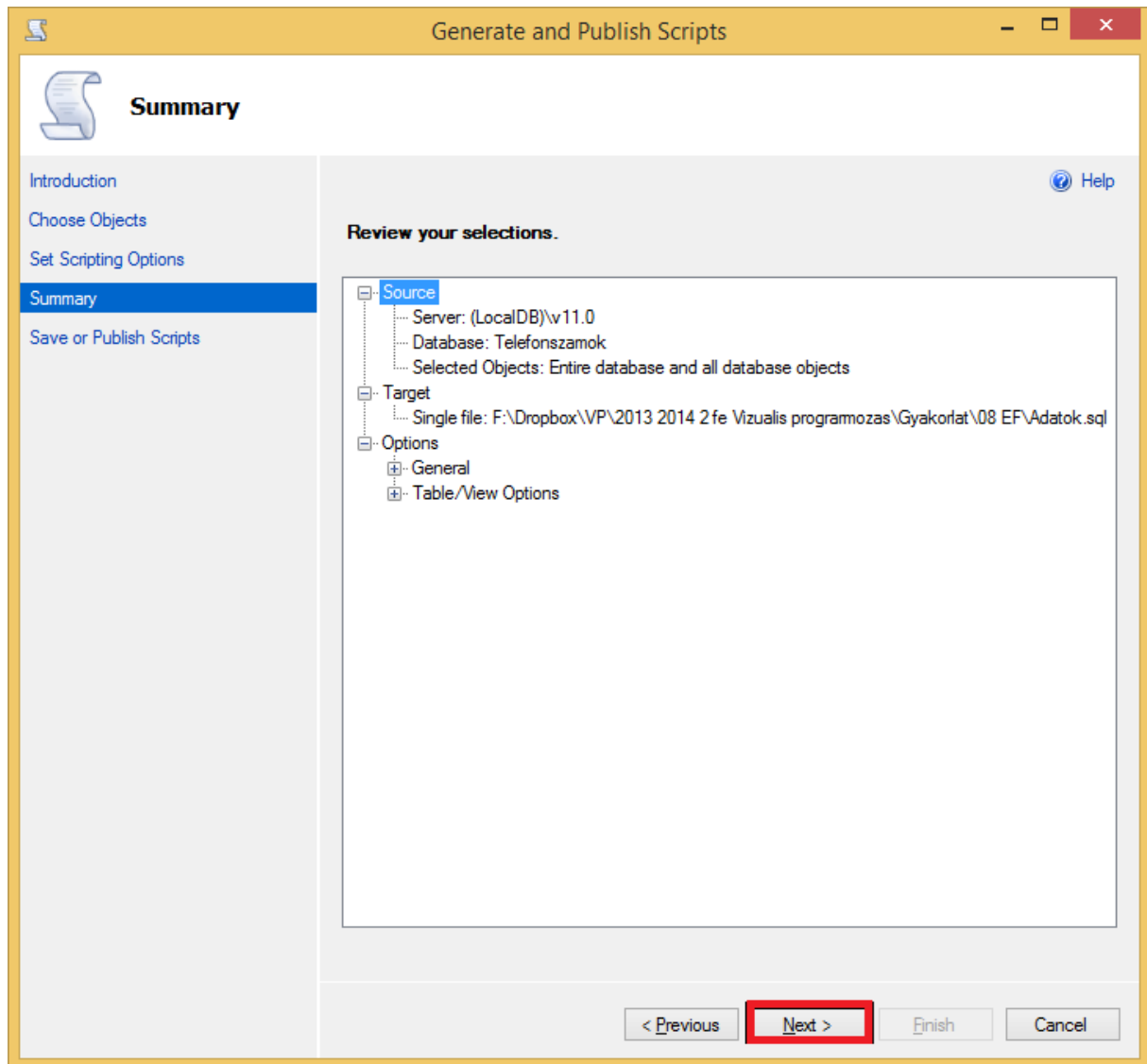
< Previous

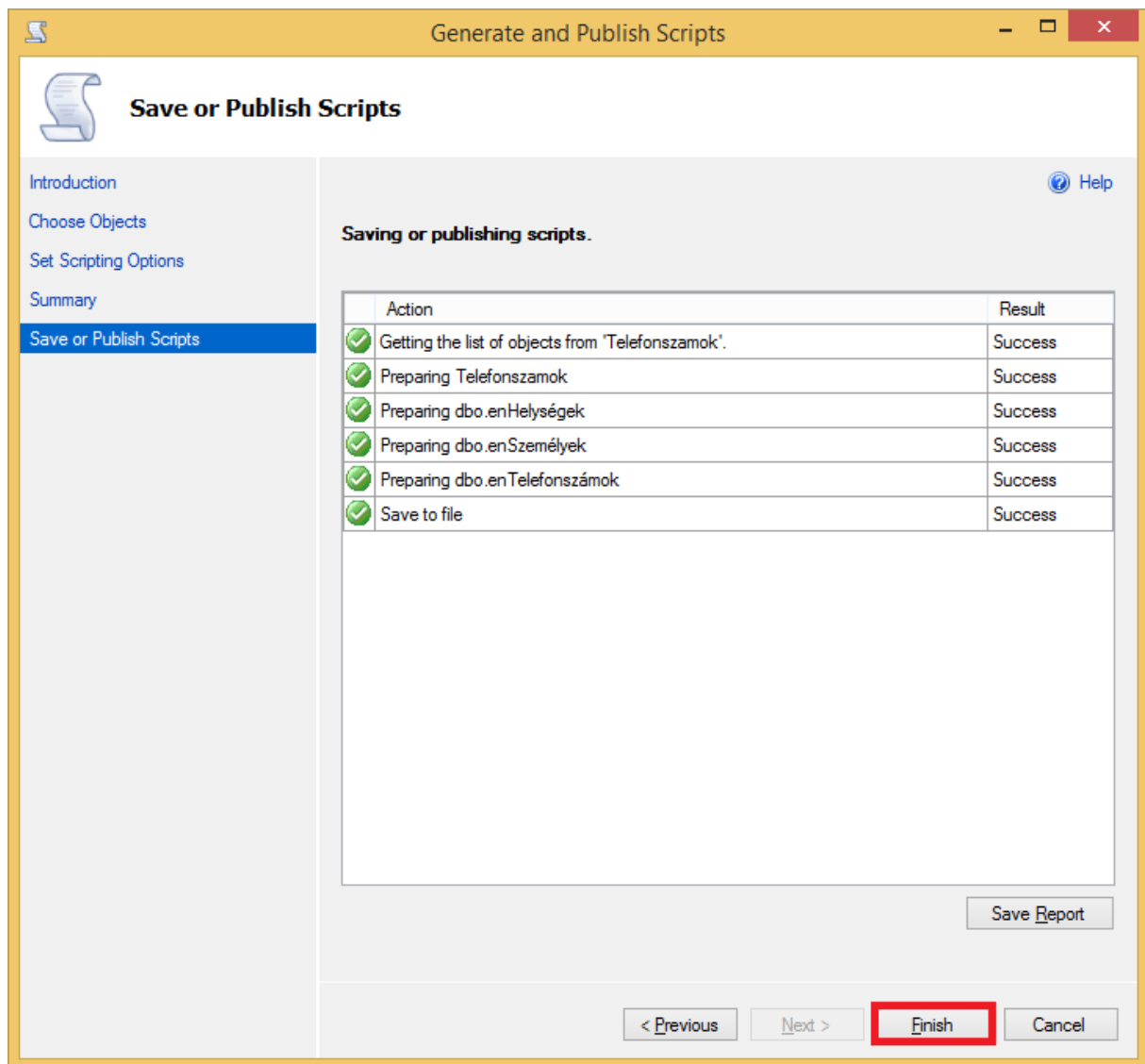
Next >

Finish

Cancel







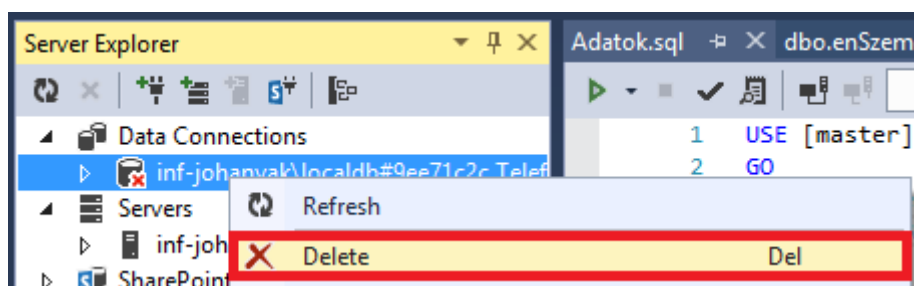
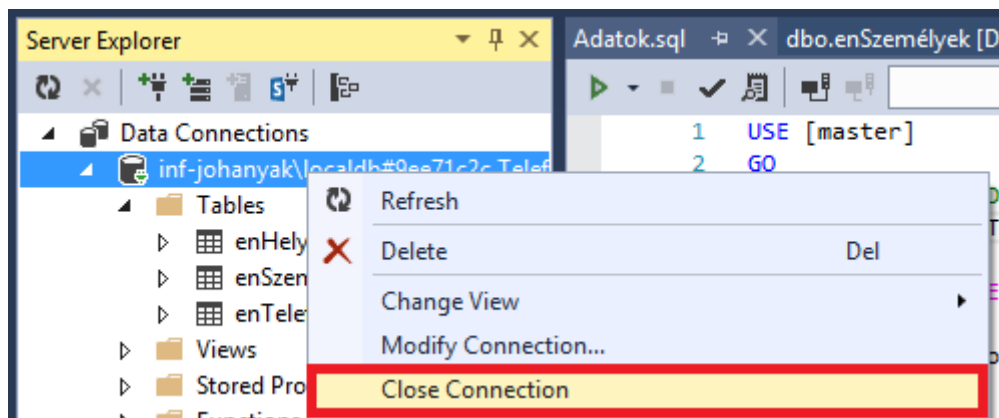
Visual Studio-ban nyissuk meg a szkriptet. Töröljük belőle az adatbázis fájl létrehozására vonatkozó részt (a kijelölt rész az alábbi képen), majd mentjük el a szkriptet.

```

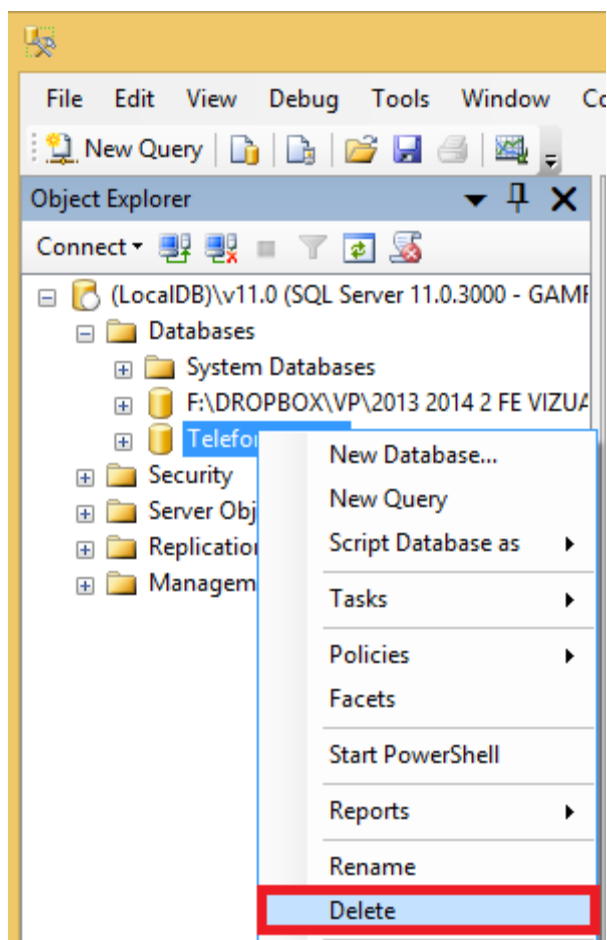
Adatok.sql  X  dbo.enSzemelyek [Data]  Object Browser  dbo.enHelysegek [Data]  Program.cs*  edmTelefonsszamok.edmx [Diagram1]
1  USE [master]
2  GO
3  /***** Object: Database [Telefonsszamok]    Script Date: 04/14/2014 08:38:14 *****/
4  CREATE DATABASE [Telefonsszamok] ON PRIMARY
5  ( NAME = N'Telefonsszamok', FILENAME = N'C:\Users\johanyak.csaba\Telefonsszamok.mdf' , SIZE = 3136KB , MAXSIZE = UNLIMITED ,
6  LOG ON
7  ( NAME = N'Telefonsszamok_log', FILENAME = N'C:\Users\johanyak.csaba\Telefonsszamok_log.ldf' , SIZE = 784KB , MAXSIZE =

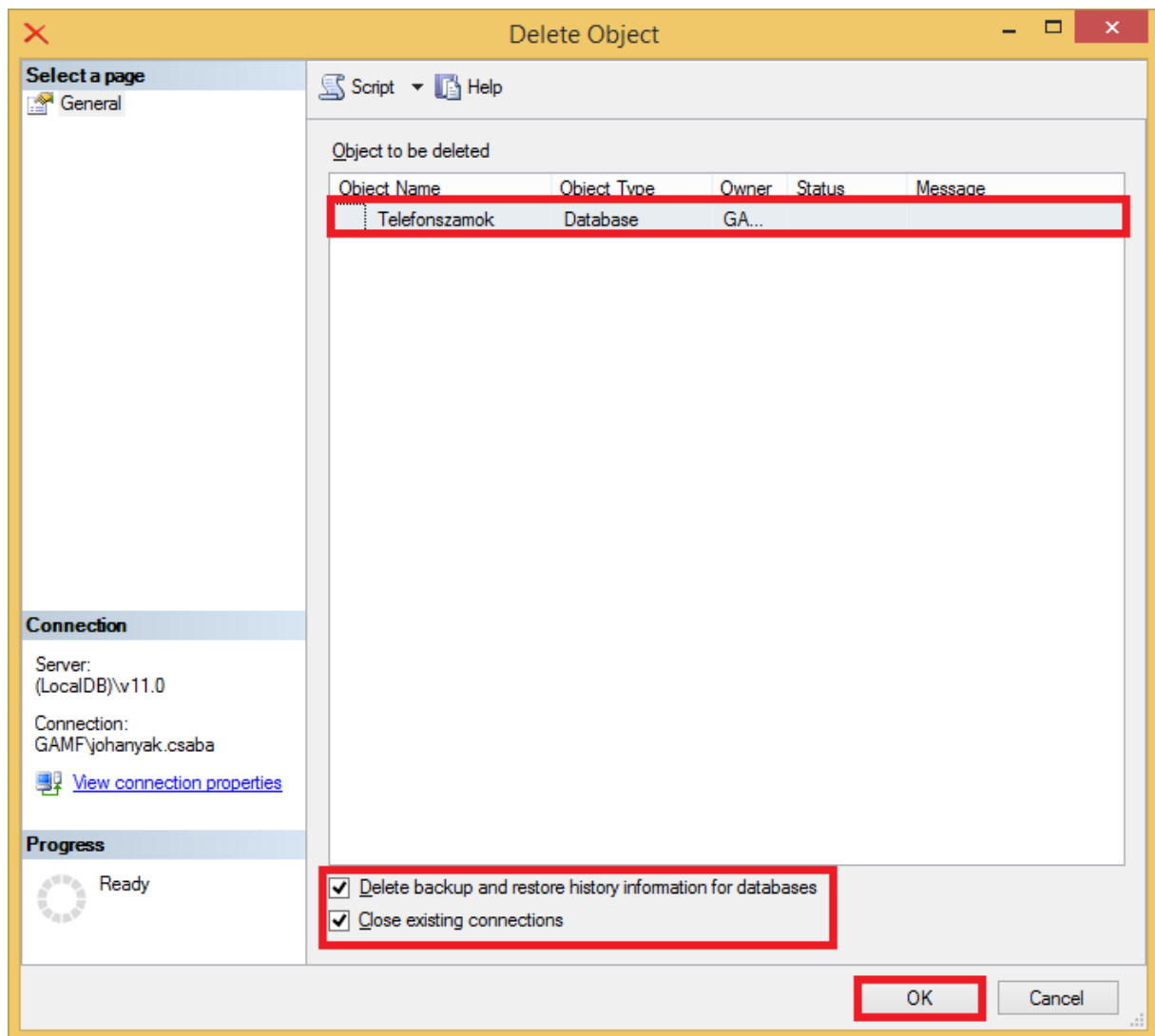
```

A szkript kipróbálásához először töröljük a meglévő adatbázist. Ehhez zárjuk le Visual Studioban a kapcsolatot, majd töröljük a kapcsolatot.

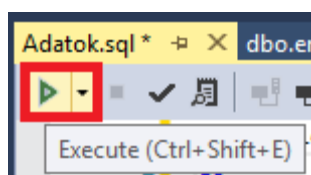


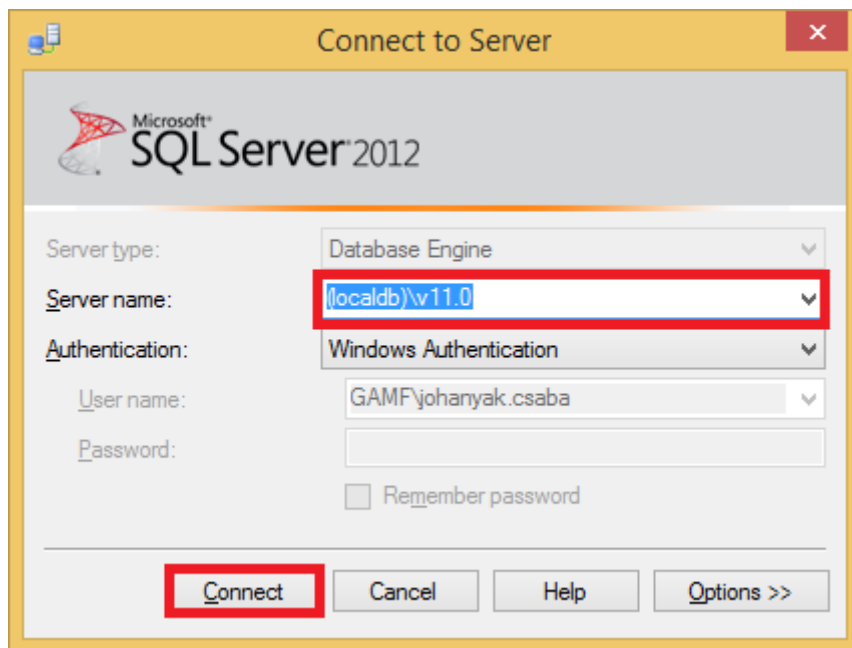
Microsoft SQL Management Studio-ban töröljük az adatbázist.





Futtassuk le a szkriptet.

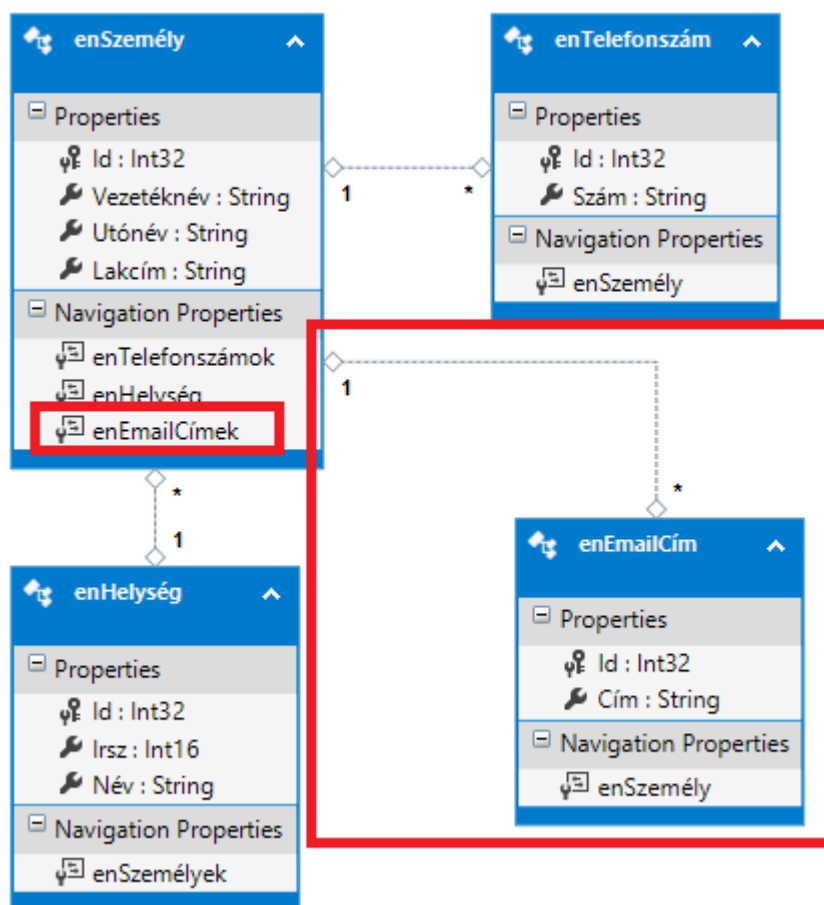


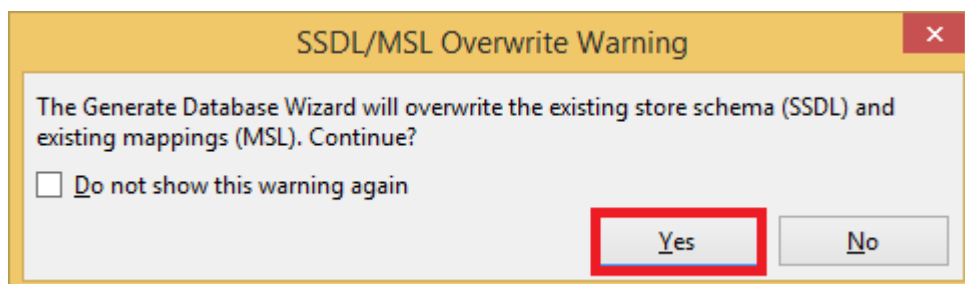
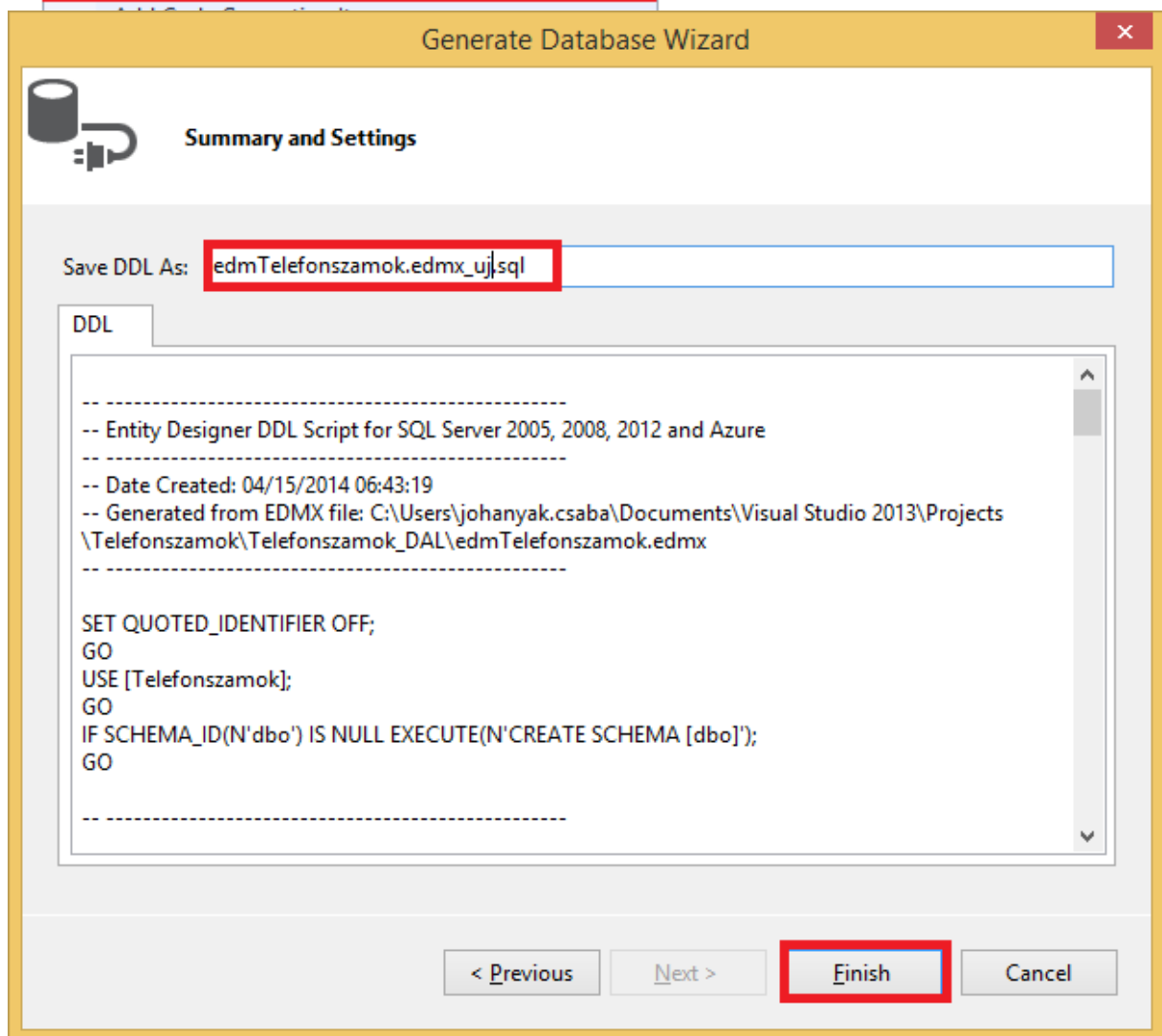
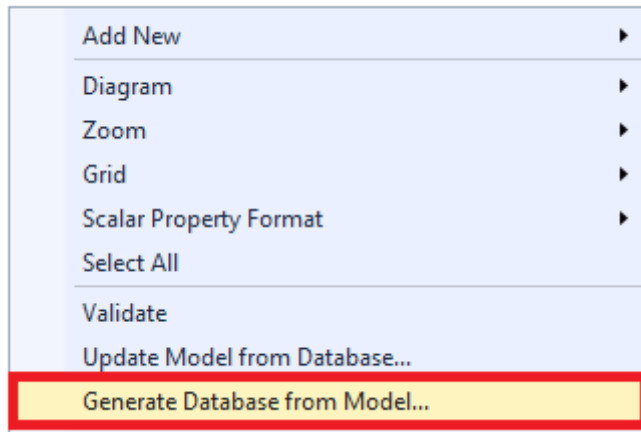


Ellenőrizzük le az adatbázis meglétét a Microsoft SQL Management Studio-ban.

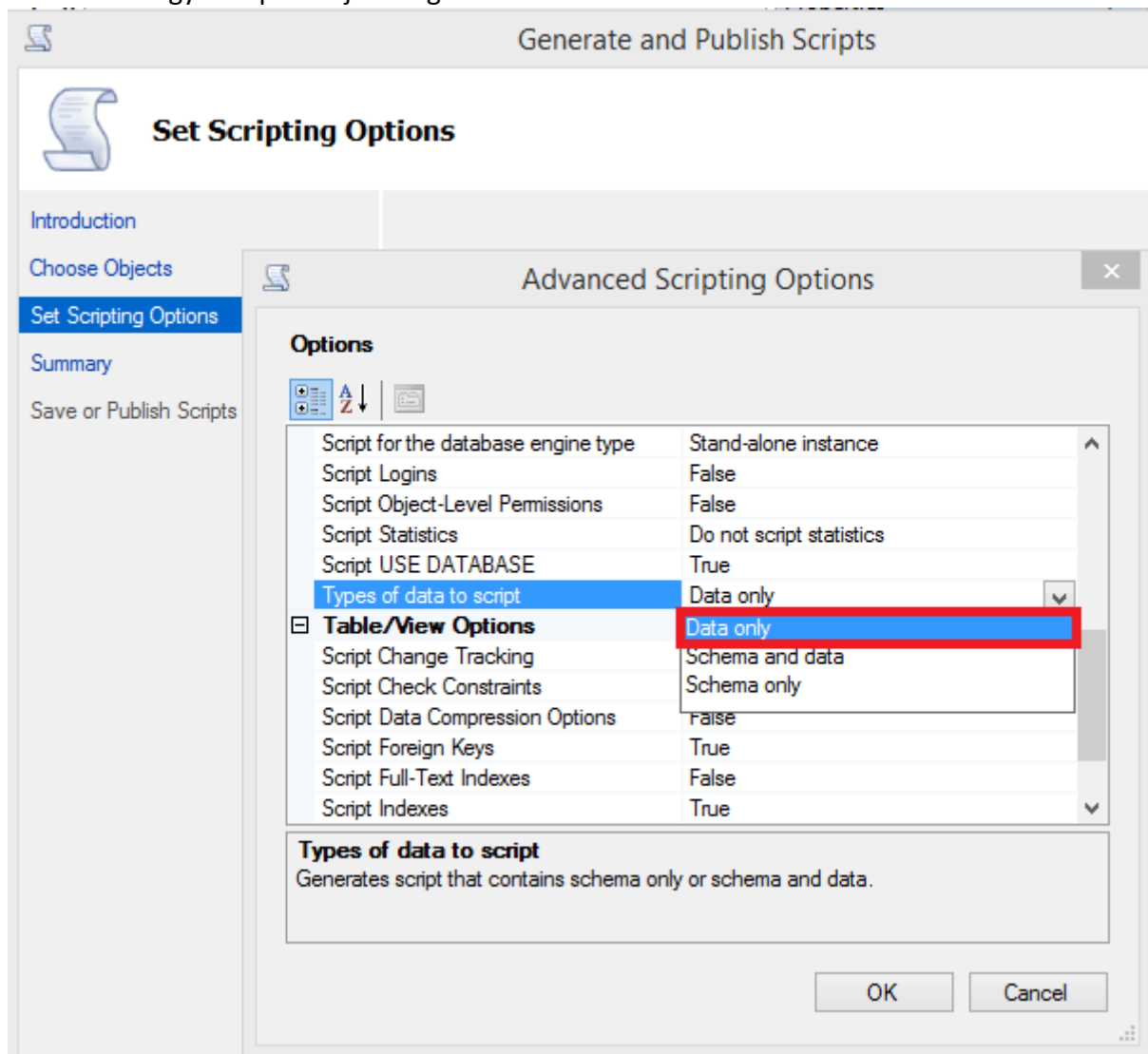
6. Adatbázis bővítése újabb entitással

Hozunk létre egy újabb entitást a diagramban az e-mail címek tárolására. Egy személynek több e-mail címe is lehet.





Mentsük le egy szkriptbe a jelenlegi adatbázis adatait.



Futtassuk le az edmTelefonszamok.edmx_uj.sql szkriptet, majd ellenőrizzük le az új adatbázis létrejöttét.

```

Csak_Adatok.sql  edmTelefonaszamok.edmx_uj.sql  Object Browser  Adatok.sql  Program.cs  edmTelefonaszamok.edmx [Diagram1]*
1  USE [Telefonaszamok]
2  GO
3  /***** Object: Table [dbo].[enHelysegiek]    Script Date: 04/15/2014 06:50:10 *****/
4  SET IDENTITY_INSERT [dbo].[enHelysegiek] ON
5  INSERT [dbo].[enHelysegiek] ([Id], [Irsz], [Név]) VALUES (1, 6000, N'Kecskemét')
6  INSERT [dbo].[enHelysegiek] ([Id], [Irsz], [Név]) VALUES (2, 2038, N'Sóskút')
7  INSERT [dbo].[enHelysegiek] ([Id], [Irsz], [Név]) VALUES (3, 2039, N'Pusztaszámor')
8  INSERT [dbo].[enHelysegiek] ([Id], [Irsz], [Név]) VALUES (4, 2040, N'Budaörs')
9  INSERT [dbo].[enHelysegiek] ([Id], [Irsz], [Név]) VALUES (5, 2045, N'Törökbálint')
10 INSERT [dbo].[enHelysegiek] ([Id], [Irsz], [Név]) VALUES (6, 2090, N'Remeteszőlős')
11 SET IDENTITY_INSERT [dbo].[enHelysegiek] OFF
12 /***** Object: Table [dbo].[enSzemelyek]    Script Date: 04/15/2014 06:50:10 *****/
13 SET IDENTITY_INSERT [dbo].[enSzemelyek] ON
14 INSERT [dbo].[enSzemelyek] ([Id], [Vezetéknév], [Utónév], [Lakcím], [enHelyseg_Id]) VALUES (1, N'Senki', N'Alfonz', N'Kis út 1.', 1)
15 INSERT [dbo].[enSzemelyek] ([Id], [Vezetéknév], [Utónév], [Lakcím], [enHelyseg_Id]) VALUES (4, N'Gipsz', N'Jakab', N'Malom köz 1.', 1)
16 INSERT [dbo].[enSzemelyek] ([Id], [Vezetéknév], [Utónév], [Lakcím], [enHelyseg_Id]) VALUES (5, N'Erős', N'Áron', N'Alma rét 3.', 2)
17 INSERT [dbo].[enSzemelyek] ([Id], [Vezetéknév], [Utónév], [Lakcím], [enHelyseg_Id]) VALUES (6, N'Olvasó', N'Jolán', N'Hermelin sugárút 13.', 3)
18 INSERT [dbo].[enSzemelyek] ([Id], [Vezetéknév], [Utónév], [Lakcím], [enHelyseg_Id]) VALUES (7, N'Argon', N'Géza', N'Ordas köz 6', 6)
19 SET IDENTITY_INSERT [dbo].[enSzemelyek] OFF
20 /***** Object: Table [dbo].[enTelefonaszamok]    Script Date: 04/15/2014 06:50:10 *****/
21 SET IDENTITY_INSERT [dbo].[enTelefonaszamok] ON
22 INSERT [dbo].[enTelefonaszamok] ([Id], [Szám], [enSzemely_Id]) VALUES (1, N'+36-30-555-5555', 1)
23 INSERT [dbo].[enTelefonaszamok] ([Id], [Szám], [enSzemely_Id]) VALUES (2, N'+36-70-456-7891', 1)
24 INSERT [dbo].[enTelefonaszamok] ([Id], [Szám], [enSzemely_Id]) VALUES (4, N'+36-20-111-1111', 4)
25 INSERT [dbo].[enTelefonaszamok] ([Id], [Szám], [enSzemely_Id]) VALUES (5, N'+36-20-222-2222', 6)
26 INSERT [dbo].[enTelefonaszamok] ([Id], [Szám], [enSzemely_Id]) VALUES (6, N'+36-76-555-555', 7)
27 INSERT [dbo].[enTelefonaszamok] ([Id], [Szám], [enSzemely_Id]) VALUES (7, N'+36-30-556-555', 7)
28 SET IDENTITY_INSERT [dbo].[enTelefonaszamok] OFF
29

```

Futtassuk le az adatokat tartalmazó szkriptet. Vigyünk fel néhány e-mail címet Visual Studio-ból.

dbo.enEmailCimek [Data] X dbo.enSzemelyek [Data] Csak_A			
Max Rows: 1000			
	Id	Cím	enSzemely_Id
	1	sea@m.hu	1
	2	senki@citromail.hu	1
	3	gj@m.hu	4
	4	erar@m.hu	5
	6	oljo@m.hu	6

Egészítsük ki a lekérdez metódust úgy, hogy az e-mail címeket is jelenítse meg.

```

private static void Lekérdez()
{
    Console.WriteLine("Összes adat\r\n-----");
    foreach (var x in cnTelefonaszamok.enSzemelyek)
    {
        var s = x.Vezetéknév + " " + x.Utónév + ": " + x.enHelyseg.Irsz +
" " +
            x.enHelyseg.Név + ", " + x.Lakcím + ", ";
        foreach (var y in x.enTelefonaszamok)
        {
            s += y.Szám+", ";
        }
        foreach (var y in x.enEmailCimek)
        {
            s += y.Cím;
            if (y != x.enEmailCimek.Last())
                s += ", ";
        }
        Console.WriteLine(s);
    }
}

```

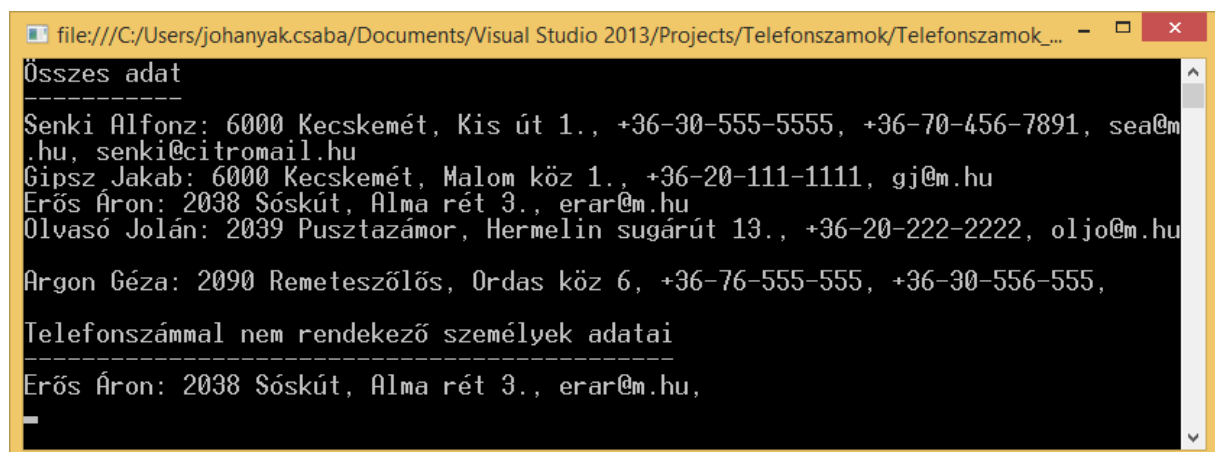
Mivel a telefonszám után most már mindig kell vessző, ezért a telefonszámos foreach ciklus magja egyszerűsödött, sőt lecserélhetjük egy LINQ kifejezésre.

```
s = x.enTelefonszámok.Aggregate(s, (current, y) => current + (y.Szám + ", "));
```

Írassuk ki a telefonszámmal nem rendelkező személyek adatait.

```
Console.WriteLine();
Console.WriteLine("Telefonszámmal nem rendelkező személyek adatai");
Console.WriteLine("-----");
var er = (from x in cnTelefonszámok.enSzemélyek
         where x.enTelefonszámok.Count == 0
         select x).ToList();

var z = er.Aggregate("", (current, x) => current + x.Vezetéknév +
    " " + x.Utónév + ": " +
    x.enHelység.Irsz + " " +
    x.enHelység.Név + ", " + x.Lakcím + ", " +
    x.enEmailCímek.Aggregate("", (curr, y) => curr +
        (y.Cím + ", ")));
Console.WriteLine(z);
```



```
file:///C:/Users/johanyak.csaba/Documents/Visual Studio 2013/Projects/Telefonszamok/Telefonszamok_...
Összes adat
-----
Senki Alfonz: 6000 Kecskemét, Kis út 1., +36-30-555-5555, +36-70-456-7891, sea@m
.hu, senki@citromail.hu
Gipsz Jakab: 6000 Kecskemét, Malom köz 1., +36-20-111-1111, gj@m.hu
Erős Áron: 2038 Sósút, Alma rét 3., erar@m.hu
Olvasó Jolán: 2039 Pusztazámor, Hermelin sugárút 13., +36-20-222-2222, oljo@m.hu
Argon Géza: 2090 Remeteszőlős, Ordas köz 6, +36-76-555-555, +36-30-556-555,
Telefonszámmal nem rendelkező személyek adatai
-----
Erős Áron: 2038 Sósút, Alma rét 3., erar@m.hu,
```

Módosítsuk úgy a sztring összeállítását, hogy az utolsó e-mail cím után ne legyen vessző.

```
var z = er.Aggregate("", (current, x) => current + x.Vezetéknév +
    " " + x.Utónév + ": " +
    x.enHelység.Irsz + " " +
    x.enHelység.Név + ", " + x.Lakcím + ", " +
    x.enEmailCímek.Aggregate("", (curr, y) =>
        (y!=x.enEmailCímek.Last())?curr +
        (y.Cím + ", "):y.Cím));
```

Tartalomjegyzék

Entity Framework alapú adatbáziselés	1
--------------------------------------	---

1. Az Entity Framework modell és az adatbázis létrehozása	1
2. Adatfelvitel közvetlenül.....	15
3. Adatfelvitel programból	16
4. Lekérdezések programból	22
5. Az adatbázisban tárolt adatok lementése egy SQL szkriptbe az adatbázis szerkezettel együtt, majd az adatbázis újbóli létrehozása.	23
6. Adatbázis bővítése újabb entitással	33
Tartalomjegyzék	37