

Szoftvertchnológia 7. gyakorlat

Román Gábor

1. Feladat

Készítsen az alábbi leírás alapján egy **Egyed kapcsolat (EK)** diagramot, majd annak alapján Visual Studio 2013-ban **Entity Framework** entitás diagramot!

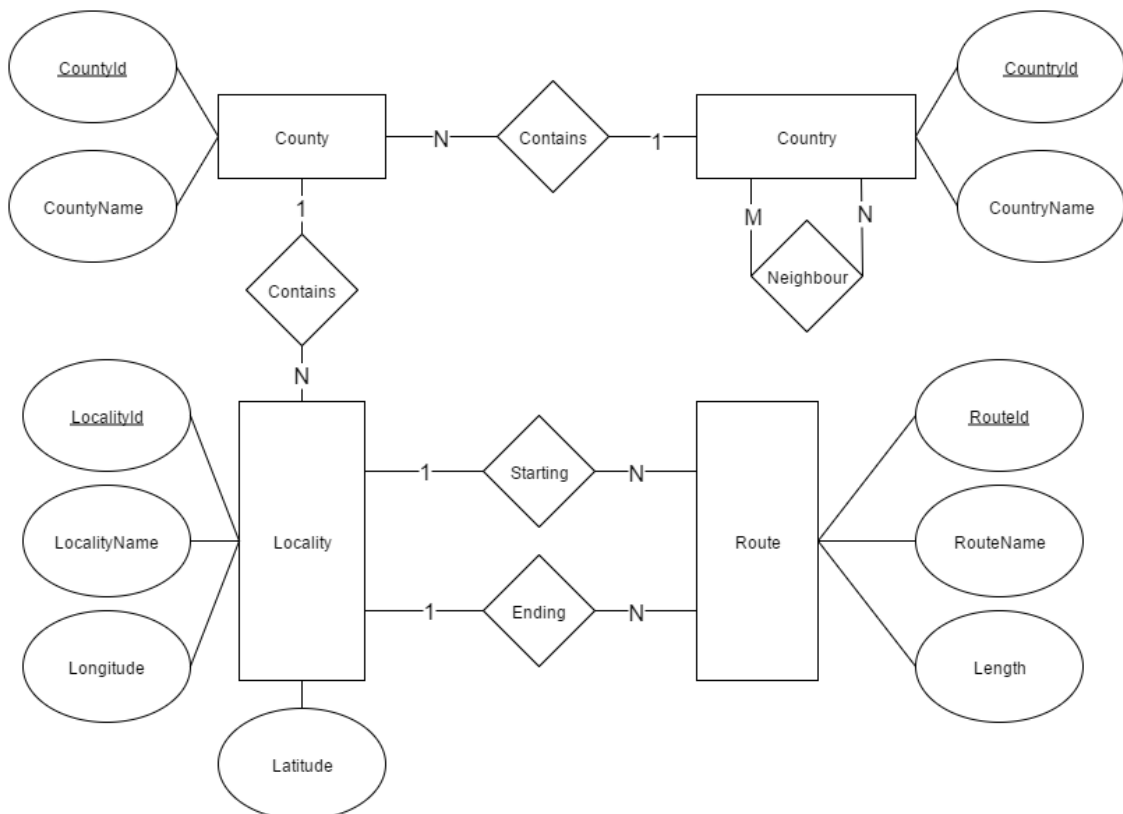
Leírás

Földrajzi adatbázisban tárolunk Országokat (Country), Megyéket (County), Helységeket (Locality) és Utakat (Route). Minden objektum rendelkezik egyedi azonosítóval (Id) és névvel (Name). Minden helységhez megadjuk a szélességi (Latitude) és hosszúsági (Longitude) koordinátáit, továbbá az utakhoz megadjuk azok hosszát (Length) kilométerben.

Egy országhoz több megye, egy megyéhez pedig több helység tartozhat. Egy helységből több út is kiindulhat, de egy út kizárólag két helységet köthet össze. (Feltesszük, hogy nincsenek „hurok utak”.) Így két kapcsolat lesz a helységek és az utak között. Az egyik a kiindulópontját, a másik a végpontját fogja szimbolizálni az útnak. Az országok határosak egymással, egy több-több kapcsolattal megvalósítva (Neighbours).

2. EK diagram

Az EK diagram elkészíthető papíron ceruzával. Az alábbi ábrán egy minta megoldás látható. A feladat az, hogy a hallgatói csoportok önállóan készítsenek egy EK diagramot, amit aztán megbeszélnek a gyakorlatvezetővel.



3. Entity Framework diagram

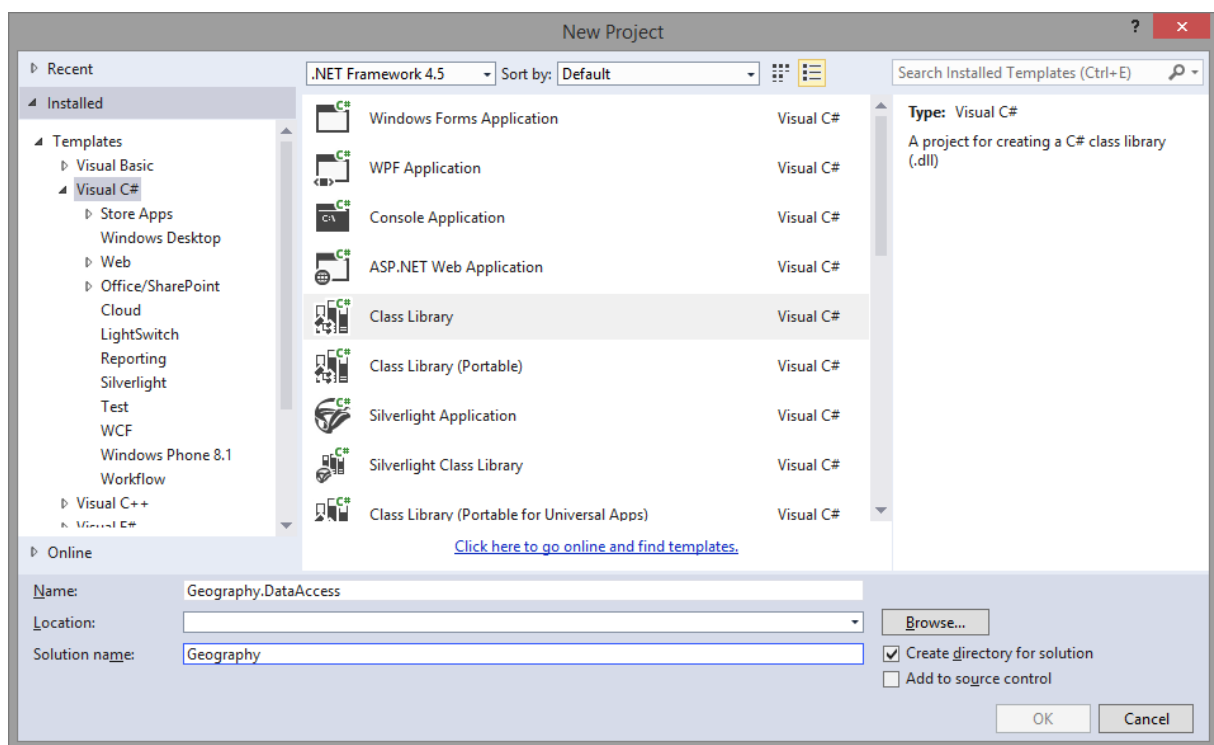
Az alábbiakban ismertetésre kerülő EF diagram az előző szakaszban ismertetett mintamegoldás alapján készült.

Típusok

- Az egyedi azonosítók elsődleges kulcsok és **Int32** típusúak.
- A szélességi és hosszúsági koordináták típusa **Double**.
- Az utak hosszának **Double** a típusa.
- A nevek **String** típusúak.

Projekt létrehozása

- hozzunk létre egy **Class Library** típus projektet **Geography.DataAccess** néven, solution neve pedig legyen **Geography**.
- töröljük a Class1.cs állományt a Solution Explorer használatával.

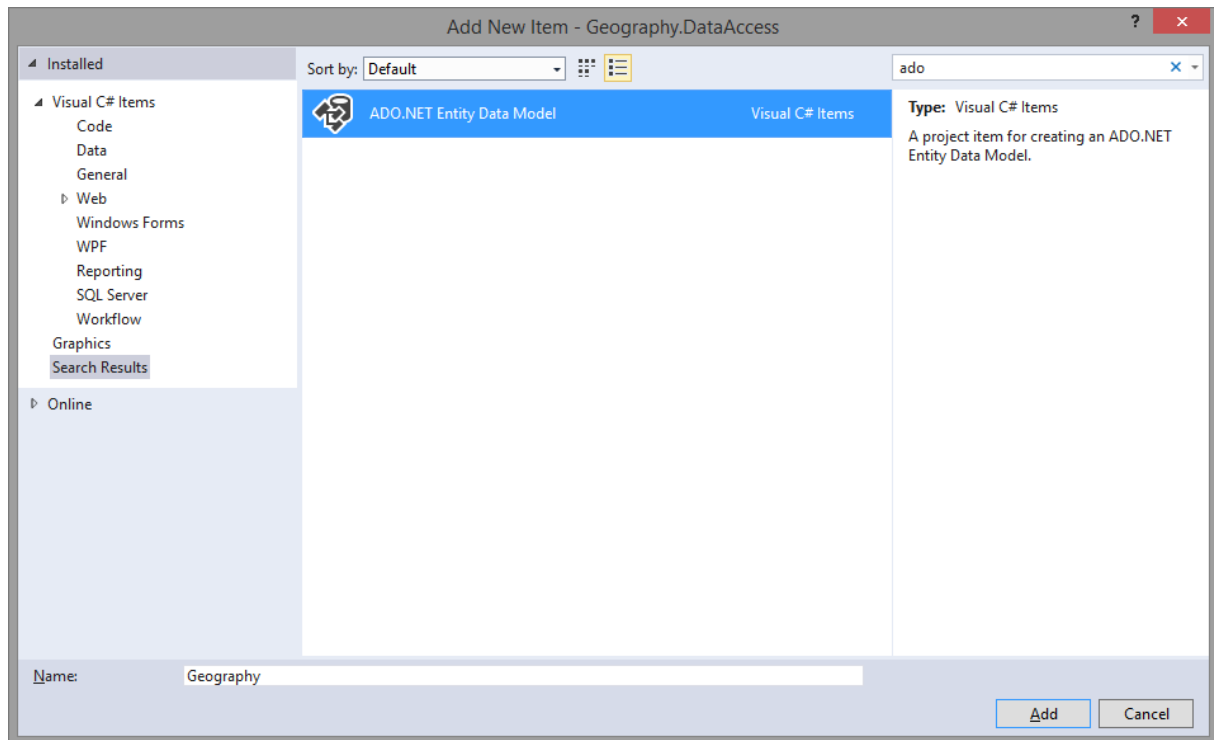


Entity framework hozzáadása a projekthez

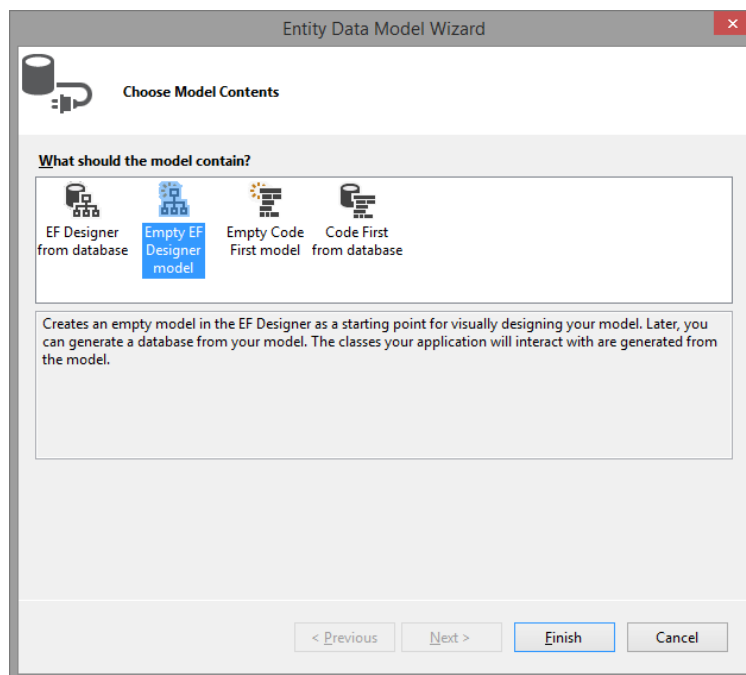
- Kattintsunk jobb egérgombbal a Geography.DataAccess projekt References mappáján, majd **Manage NuGet Packages** (Ha nincs ilyen lehetőség, akkor a Toos menü Extensions and Updates menüjében keressük ki a NuGet Package Managert és telepítjük fel.)
- NuGet Package manager-ben keressük ki (Online kategória bal oldalt) az **EntityFramework**-öt, majd install.

Entitás modell létrehozása

- Adjuk hozzá az entitás modellt
 - o jobb klikk a projekten, Add, New Item, a keresőmezőbe írjuk be hogy „ado”
 - o keressük ki az **ADO.NET Entity Data Model**-t
 - o neve legyen **Geography**

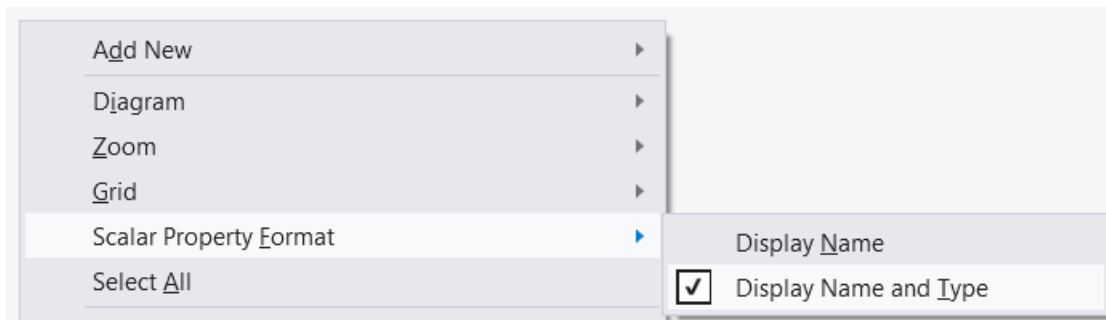


- válasszuk a **Empty model-t** (Code first), majd Finish



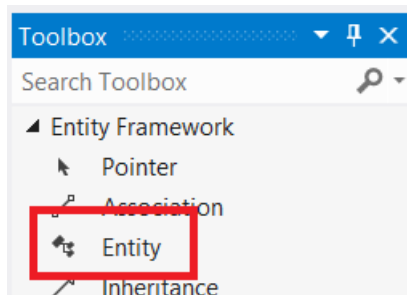
Entitás osztályok hozzáadása

- állítsuk be, hogy ne csak a tulajdonság neveket, hanem a típusokat is mutassa
 - o jobb klikk az üres felületen -> **Scalar Property Format** -> **Display Name and Type**



Country entitás

- toolbox segítségével húzzunk rá a felületre egy entitást



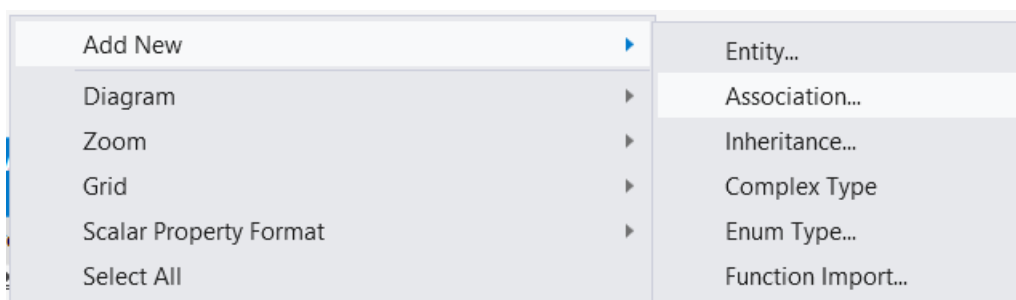
- entitás neve legyen **Country**
- entitás tulajdonságai között (jobb klikk rajta -> Properties) állítsuk be az **Entity Set Name**-et **Countries**-ra
- nevezzük át az Id-t **CountryId**-ra
- adjunk hozzá Scalar Property-eket:
 - o jobb klikk az entitáson -> Add New -> Scalar Property
 - o neve legyen **CountryName**, típus **String**

County entitás

- hozzuk létre a **County** entitás osztályt

Kapcsolat az entitások között

- jobb klikk az üres felületen -> Add New -> **Association**



- állítsuk be a kapcsolat típusát (1 és több oldalt)
 - o figyeljünk oda, hogy az **Country** entitásban a **Navigation Property Counties** nevű legyen, mivel egy országhoz több megye tartozik.

- létrejött kapcsolat:
 - o külső kulcs nevének CountryCountryId-t generál, ami nevezzük át CountryId-ra

Hozzuk létre a többi entitást az előbbieik alapján

- A **Locality** és a **Route** entitások között két egy-sok kapcsolat is lesz. Ez két azonosítót vezet be a **Route** entitásban, mind a kettő egy-egy helységnek az azonosítója. Itt adatbázis szinten külön kikötést lehet tenni, hogy a két id nem egyezhet, hiszen kizártuk az olyan út létezését, amelynek a kezdő és végpontja is ugyanaz a helység.
- Az országok határosságánál egy sok-sok kapcsolatot kell létrehozunk ugyanazon az entitáson. Ennek a kapcsolatnak a neve lehet például **Neighbours**. Ekkor nem kell megadni navigation property-t. Az adatbázis generálásnál esetén látható, hogy ennek megfelelően létrehoz egy kapcsolótáblát a program.

