# Webes és mobil programozás F#-ban 10. gyakorlat

## Berezvai Dániel jegyzete <http://elte.3ice.hu/>

Előző órán shopping cart volt, ketten voltak.

Ma tovább nézzük.

ComputeFamiliesAndProducts jó esetben adatbázisból dolgozik, Database TypeProvider segítségével.

Nincs még Persist, újratöltésnél elvesznek az adatok.

Website → Manage PAckages

FSharp.Data és FSharp.Data.TypeProvider nuget package-k telepítése.

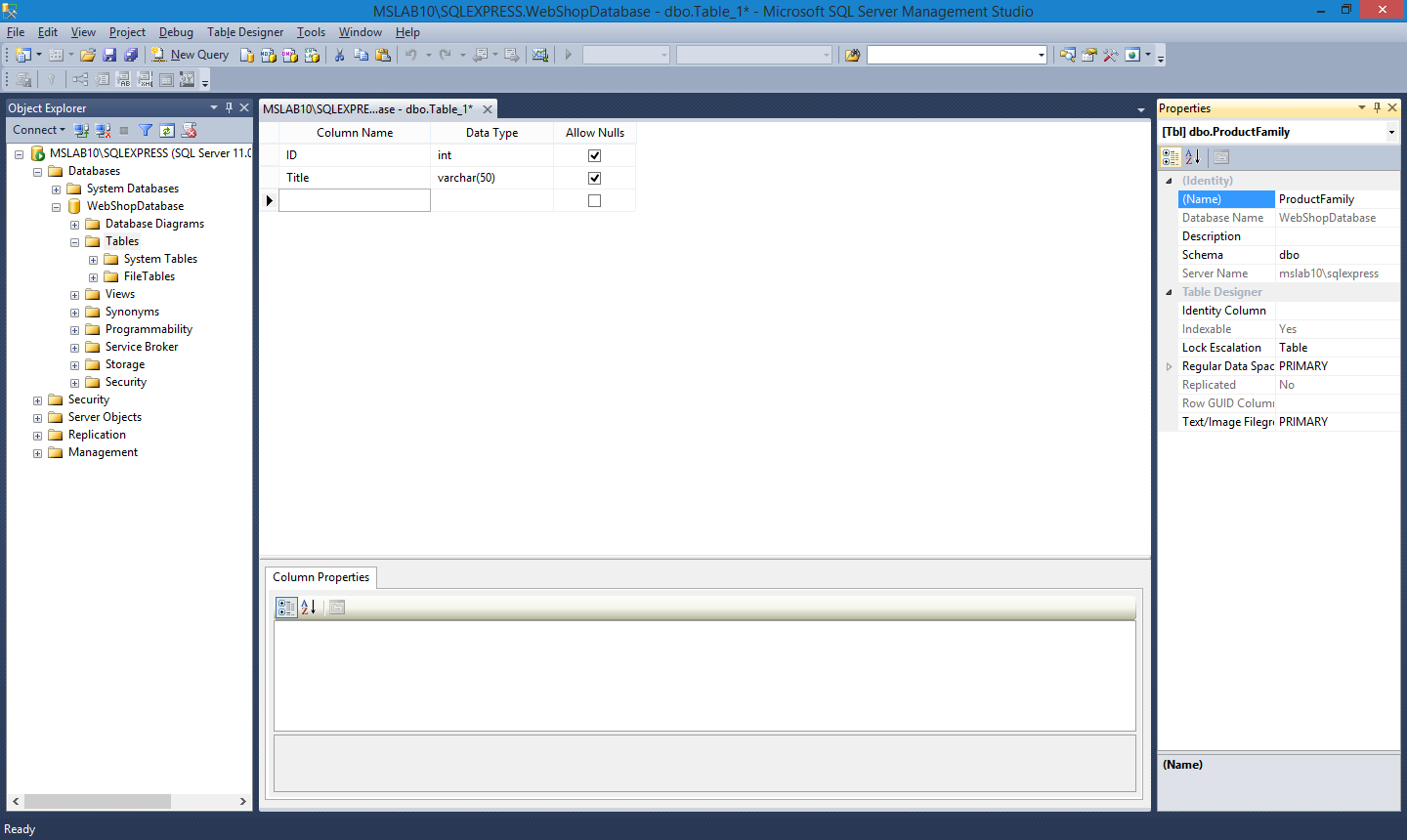
open System.Data

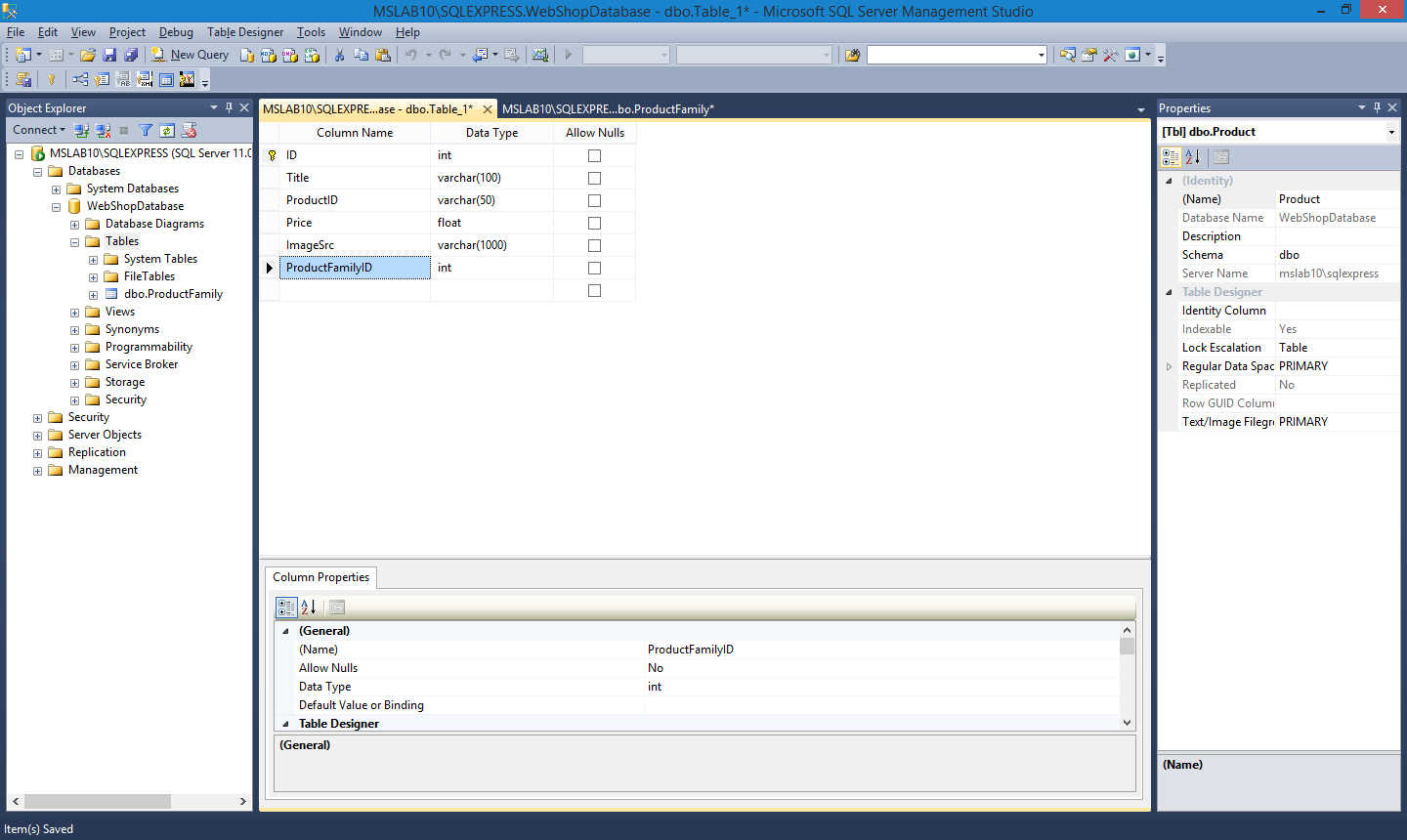
open System.Data.Linq // Ehhez kell valami reference még. System Data Linq gondolom.

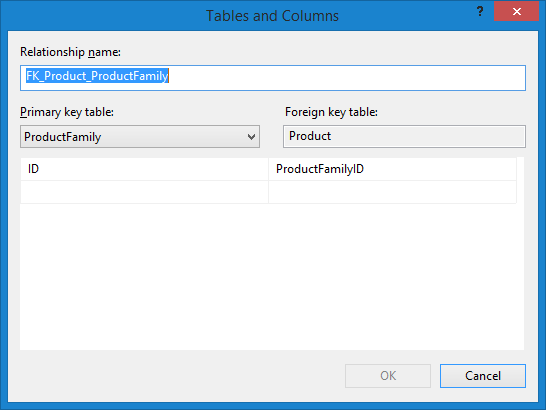
open Microsoft.FSharp.Data.TypeProviders

type dbScheme = SqlDataConnection<"Data Source=mssql1.inf.elte.hu; Initial Catalog=WebShopDatabase; Integrated Security=true">

## 3 kép







## SQL

use WebShopDatabase

go

insert WebShopDatabase.dbo.ProductFamily

values

(1,'Laptops'), (2,'Desktops'), (3,'Netboooks')

go

insert WebShopDatabase.dbo.Product

values

(1,'Toshiba','id1',1499.0,'\images\laptop.png',1),

(2,'HP,'id2',1299.0,'\images\laptop.png',1),

(3,'Dell','id3',1299.0,'\images\laptop.png',1),

(4,'Asus','id4',1199.0,'\images\laptop.png',1),

(5,'Gamer 1','id11',899.0,'\images\desktop.png',2),

(6,'Gamer 2','id11',699.0,'\images\desktop.png',2),

(7,'Office','id11',699.0,'\images\desktop.png',2),

(8,'Server','id11',1099.0,'\images\desktop.png',2),

(9,'Entry','id21',599.0,'\images\netbook.png',3),

(10,'Cool','id21',659.0,'\images\netbook.png',3),

(11,'Medium','id21',699.0,'\images\netbook.png',3),

(12,'Speed-king','id21',799.0,'\images\netbook.png',3),

go

## Program folytatása

let db = dbSchema.GetDataContext()

let products = db.Product

let productFamilyNameByID id = query{

for row in db.ProductFamily do

where (row.ID = id)

select row

} |> Seq.map (fun row → row.Title) |> Seq.head

let productListByFamilyID id =

let products = query{

for row in db.Product do

where (db.ProductFamilyID = id)

// join key in db.ProductFamily on (row.ProductFamilyID = key.ID)

select row

} |> Seq.map (fun row ->

T row.ImageSrc row.Title row.ProductID row.Price) |> Seq.toList

{CartFamily.Title = productFamilyNameByID id; CartFamily.Products = products}

let count =

query{

for row in db.ProductFamily do

select row

} |> Seq.length

let rec createProductFamilies i acc =

match count = i with

| true -> acc @ [productListByFamilyID i]

| false -> createProductFamilies (i+1) (acc @ [productListByFamilyID i])

createProductFamilies 1 []

Gyakorlat vége.