

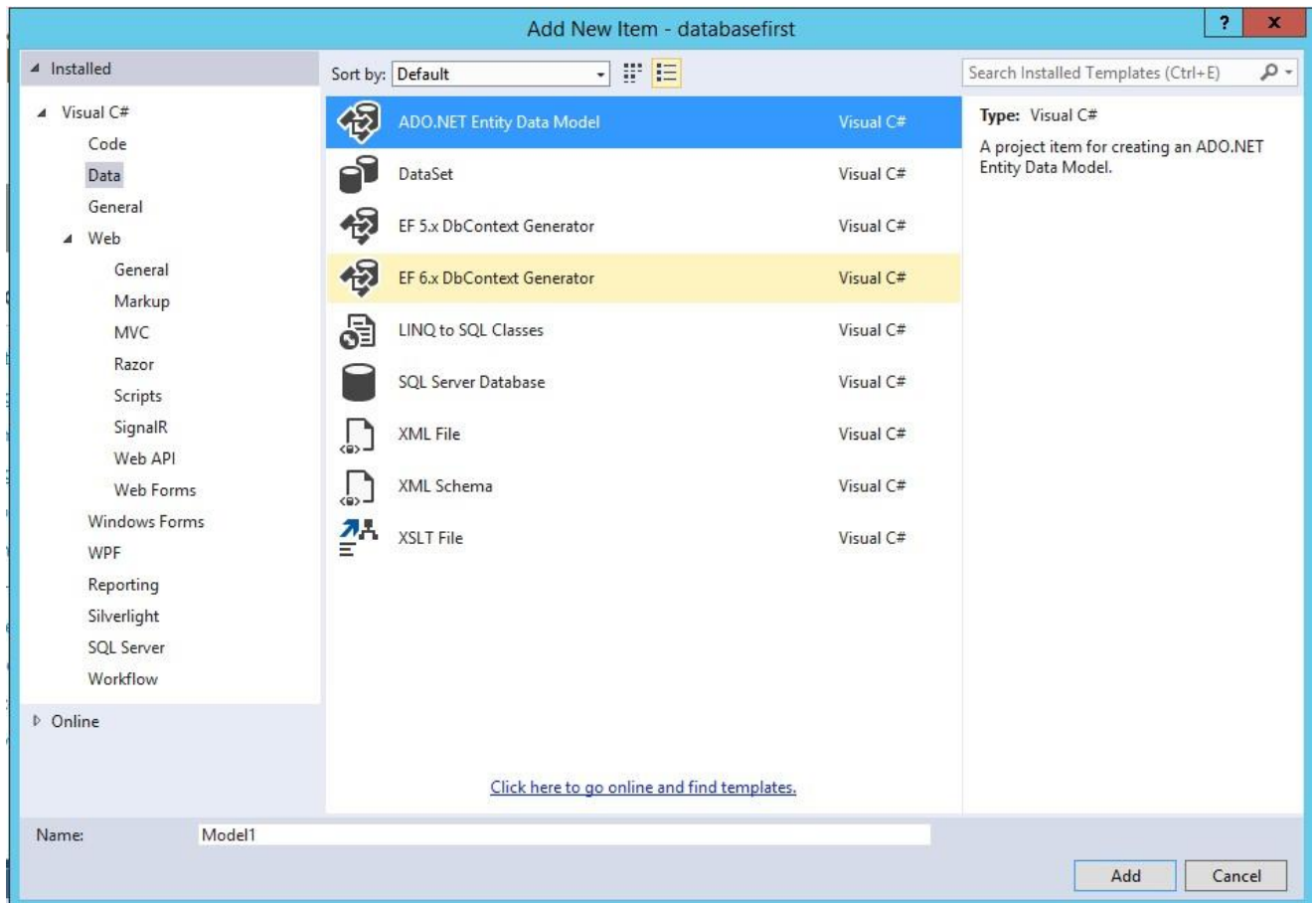


Av Erik
"Zkall"
Svensson
@esdev.se

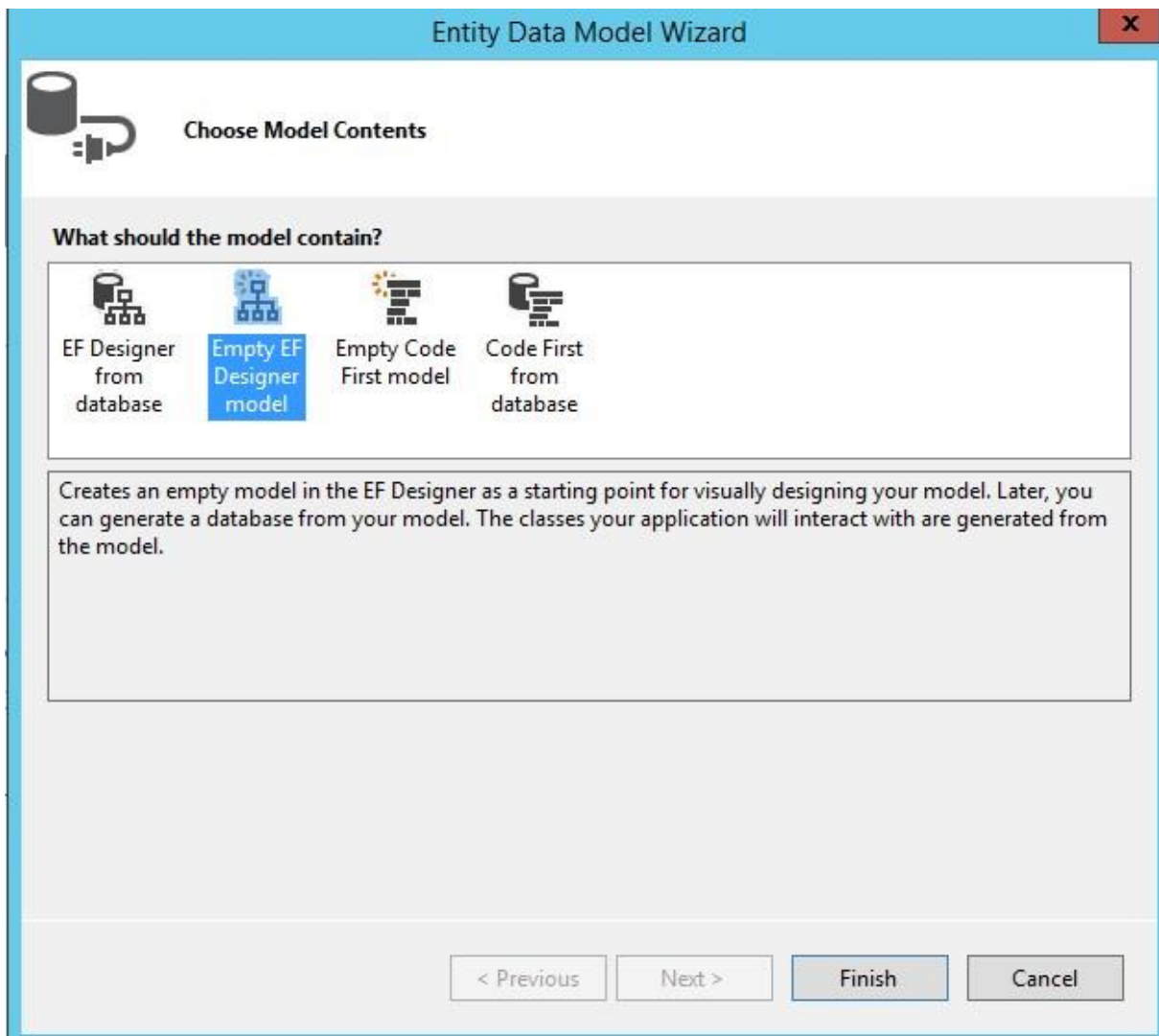


Entity Framework Designer first

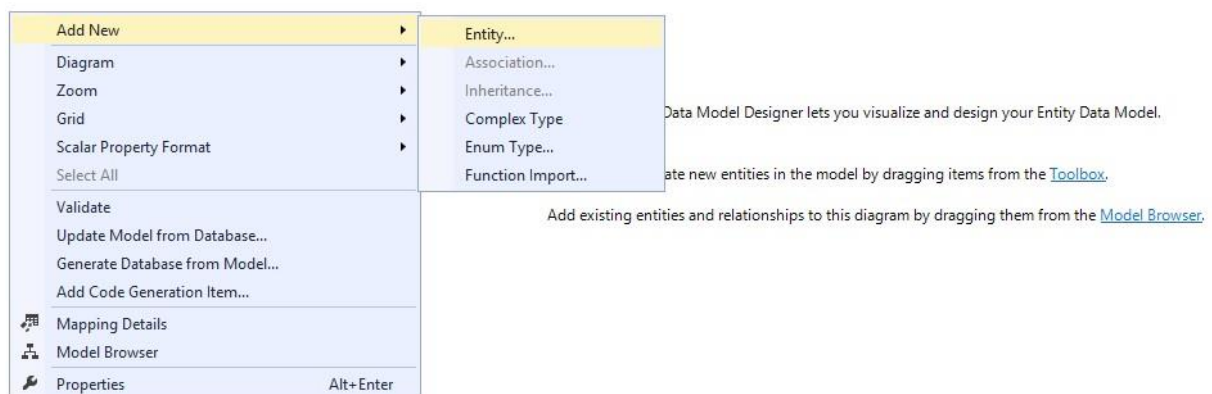
Högerklicka på models foldern, add new



Välj ADO.NET Entity Data Model och döp den till något, t.ex DB. Klicka på Add.

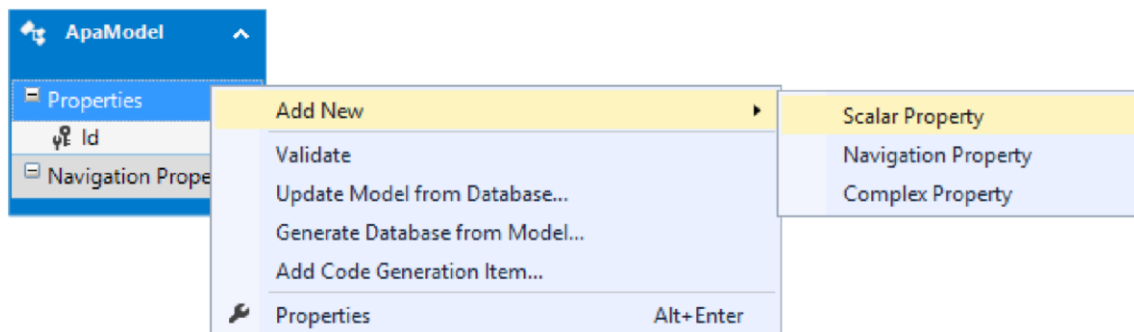


Välj empty EF Designer model och tryck finish



Högerklicka på fältet du har fått upp och välj add new Entity.

Döp din model och låt allt var ikryssat. Tryck ok.



Högerklicka på din model och tryck add new Scalar property.

Scalar property

Vanlig property på en class, kan vara int, string, etc.

Navigation property

Är när du ska navigera och redovisa för en annan entity. Till exempel så har du kanske 2 tables, apor och bananer, apor ska då länka till bananer med en navigation property.

Complex property

Se wikipedia för mer info av det, dock ej så relevant för det mesta.

Generate Database

När du är klar med att skapa alla modellerna högerklickar du på fältet och trycker på generate database from model.

Generate Database Wizard

Choose Your Data Connection

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

cookbookEntities (databasefirst) 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:

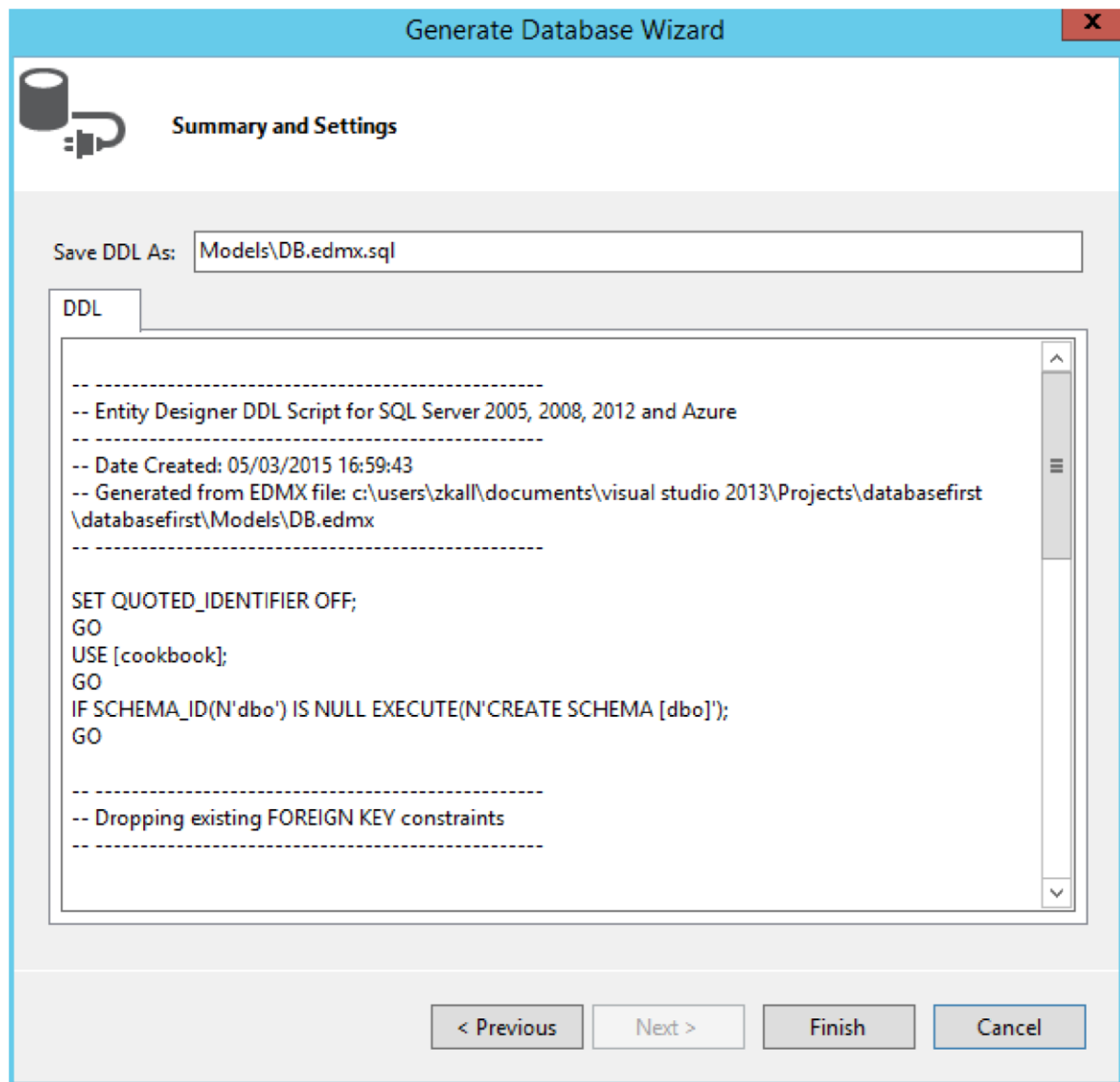
data source=db.esdev.se;initial catalog=cookbook;user id=sa;password=*****;multipleactiveresultsets=True;application name=EntityFramework

☒ Save connection settings in Web.Config as:

DBContainer

< Previous Next > Finish Cancel

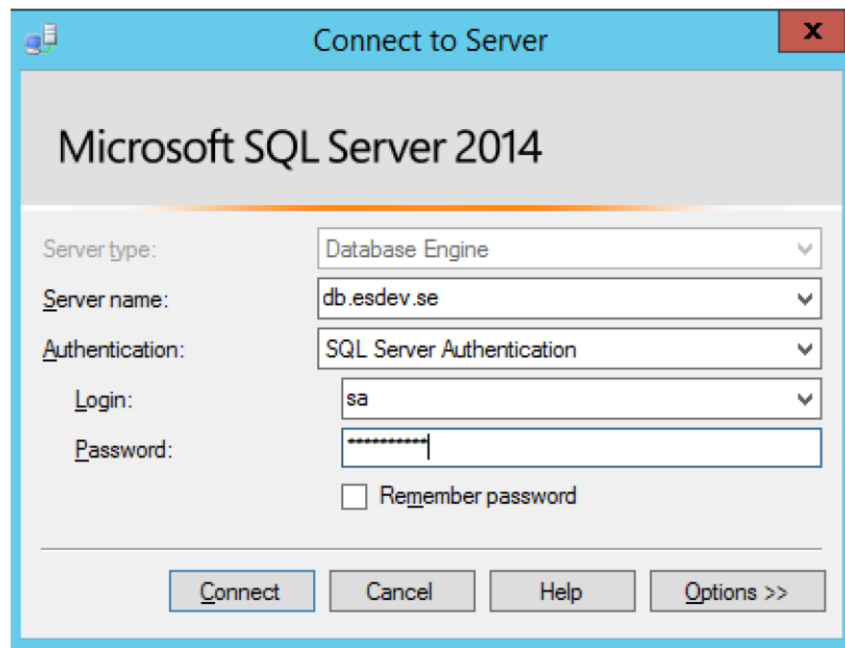
Fyll i eller använd existerande connection string, Tryck Yes, include the sensitive data in the connection string och sedan Next.



Tryck Finish. En Sql fil har nu genererats för att skapa alla databastables.



Högerklicka på db filen och tryck execute.



Fyll i din information och tryck connect.

Gratulerar, du har nu skapat din databas. Om det blev error så kolla att du använder sa kontot och har rätt lösenord, det är vanligaste problemet.

Hur kommer man åt databasen från koden

Du refererar till namnet av db classen (rekommenderat namn är DB då det är rätt uppenbart när du ska deklarerar vad som är vad. ,så du skriver:

```
private DB db = new DB();  
Oreferences  
public ActionResult Index()  
{  
    var model = db.Apor.ToList();  
    return View();  
}
```

Sedan skriver du bara db.Pluralnamn av dbset du skapade och sen db.pluralnamn.add/remote/update/

Efter du har gjort en ändring måste du skriva db.SaveChanges(); annars sparas inte ändringarna.

Lägga till en model i databasen

```
private DB db = new DB();  
Oreferences  
public ActionResult Index()  
{  
    var temp = new ApaModel { namn="Adolf" };  
    db.Apor.Add(temp);  
    db.SaveChanges();  
    return View();  
}
```

Id behöver du inte skriva, utan det sköts per automatik.

Udatera en model i databasen

```
private DB db = new DB();
0 references
public ActionResult Index()
{
    var apan = db.Apor.First(x => x.namn == "Adolf");
    apan.namn = "Nisse";
    db.SaveChanges();
    return View();
}
```

Hämta först det du ska updatara, ändra sedan variablerna du vill ändra, och avsluta med db.SaveChanges();

Ta bort en model i databasen

```
private DB db = new DB();
0 references
public ActionResult Index()
{
    var apan = db.Apor.First(x => x.namn == "Adolf");
    db.Apor.Remove(apan);
    db.SaveChanges();
    return View();
}
```

Samma som förra exemplet, hämta objektet du ska modifieraa. Skriv sedan db.Apor.Remove(apan); Avsluta med db.SaveChanges();

Hämta flera

```
private DB db = new DB();
0 references
public ActionResult Index()
{
    var model = db.Apor.ToList();
    return View();
}
```

.ToList() för List > IEnumerable imo :P

Hämta flera med parametrar

```
private DB db = new DB();
0 references
public ActionResult Index()
{
    var model = db.Apor.Where(x => x.namn == "Nisse").ToList();
    return View();
}
```

.Where(x => x.namn == "Nisse") hämtar alla apor som heter Nisse. **X => X == x.namn** är lambda som jag går in på mer i ett annat dokument, för nu kan ni tänka på att det är så det funkar bara.