

MVC 5

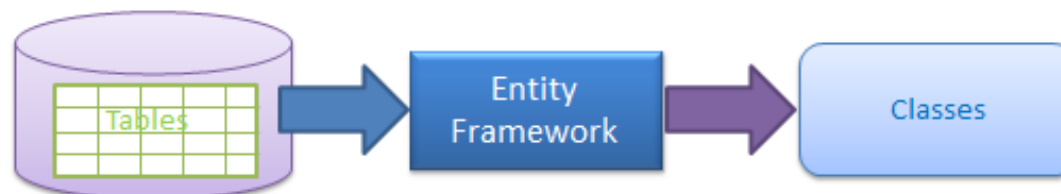
(Model-View-Controller)

Entity Framework (EF)

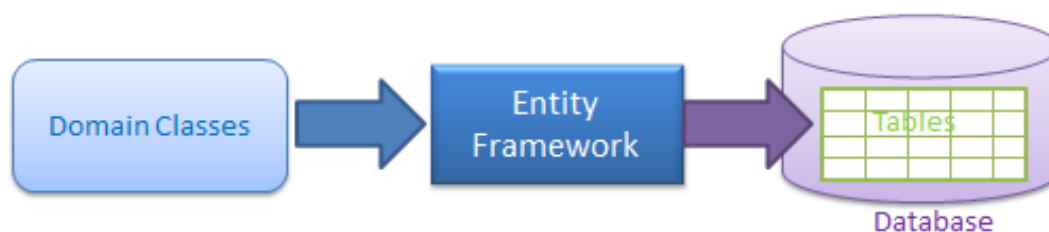
Što je Entity Framework

- O/RM (Object/Relational Mapping) radna okolina za pristup bazama podataka
- Mapiramo objekte iz baze te ih onda koristimo programirajući u .Net radnoj okolini, koristeći standardne .Net jezike (C#, Visual Basic)

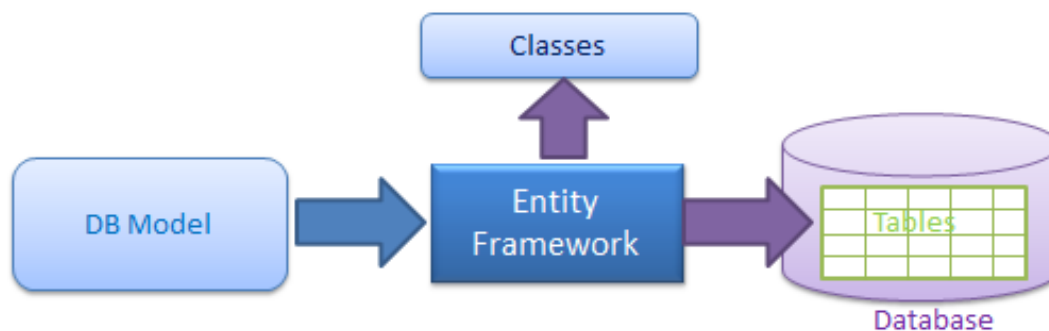
Mogućnosti EF-a



Database First - generiramo C# klase iz postojeće baze



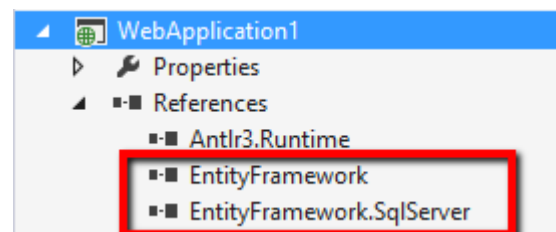
Code First - kreiramo bazu na osnovu C# klasa



Model First - kreiramo bazu i klase iz dizajniranog modela

Instaliranje EF-a

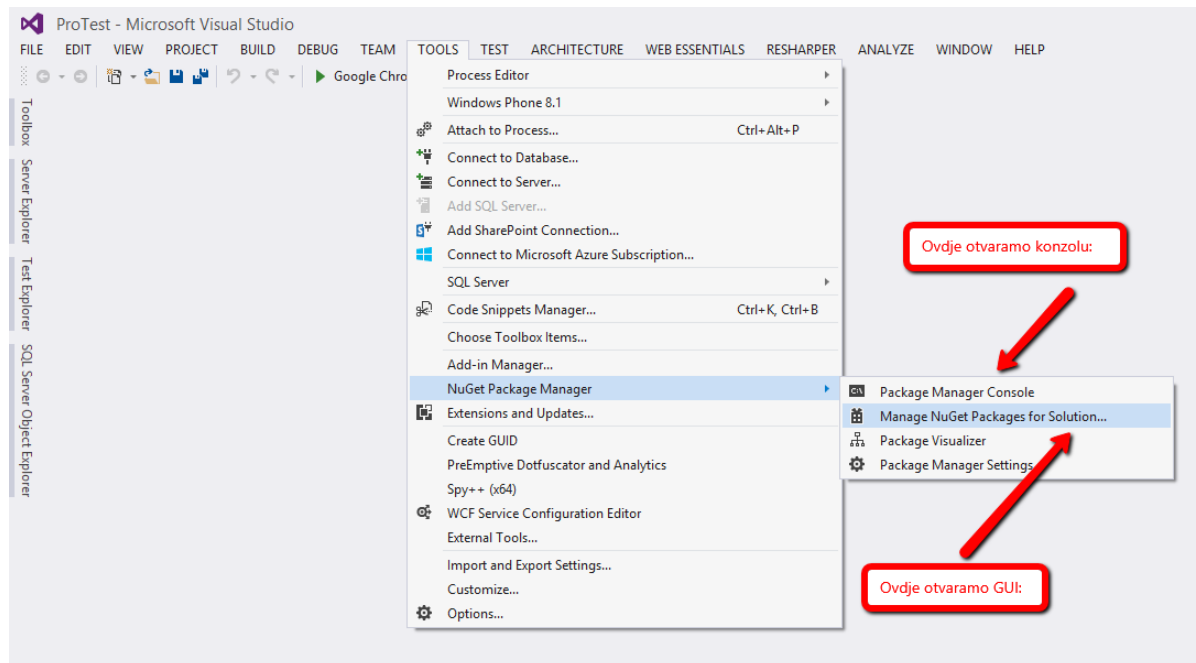
- Visual Studio 2013 defaultno instalira EF u okviru ASP.NET Web Application projektnog predloška, što možemo vidjeti po referencama novo kreiranog predloška:



- Ako želimo instalirati stariju ili noviju verziju EF-a nego što se defaultno nudi, moramo to sami napraviti

Nuget Package Manager

- Dodatak koji nam pomaže instalirati pakete poput Entity Frameworka
- Može se koristiti kroz GUI ili kroz komandnu liniju



Rad kroz komandnu liniju

■ Instalacija:

Install-Package EntityFramework

```
PM> Install-Package EntityFramework
'EntityFramework 6.1.3' already installed.
Removing 'EntityFramework 6.1.1' from WebApplication5.
Successfully removed 'EntityFramework 6.1.1' from WebApplication5.
Adding 'EntityFramework 6.1.3' to WebApplication5.
Successfully added 'EntityFramework 6.1.3' to WebApplication5.

Type 'get-help EntityFramework' to see all available Entity Framework commands.

PM> |
```

■ Deinstalacija:

Uninstall-Package EntityFramework

Package Manager Console

Package source: nuget.org Default project: TestProjektic

Each package is licensed to you by its owner. Microsoft is not responsible for, nor does it grant any licenses to, third-party packages. Some packages have licenses. Follow the package source (feed) URL to determine any dependencies.

Package Manager Console Host Version 2.8.50926.663

Type 'get-help NuGet' to see all available NuGet commands.

```
PM> Uninstall-Package EntityFramework
Uninstall-Package : Unable to uninstall 'EntityFramework 6.1.3' because 'Microsoft.AspNet.Identity.EntityFramework 2.1.0' depends on it.
At line:1 char:1
+ Uninstall-Package EntityFramework
+ ~~~~~
+ CategoryInfo          : NotSpecified: (:) [Uninstall-Package], InvalidOperationException
+ FullyQualifiedErrorId : NuGetCmdletUnhandledException,NuGet.PowerShell.Commands.UninstallPackageCommand

PM> Uninstall-Package Microsoft.AspNet.Identity.EntityFramework
Removing 'Microsoft.AspNet.Identity.EntityFramework 2.1.0' from TestProjektic.
Successfully removed 'Microsoft.AspNet.Identity.EntityFramework 2.1.0' from TestProjektic.

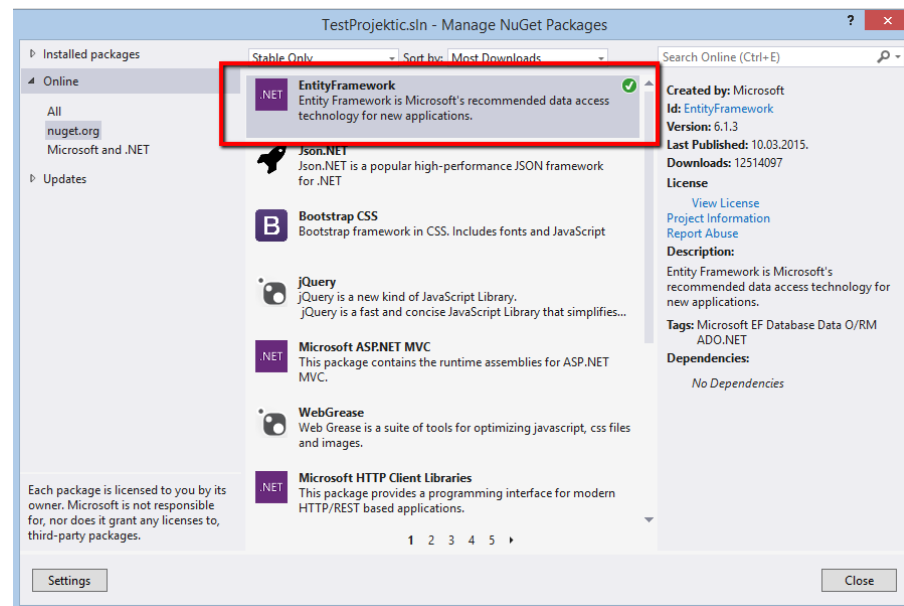
PM> Uninstall-Package EntityFramework
Removing 'EntityFramework 6.1.3' from TestProjektic.
Successfully removed 'EntityFramework 6.1.3' from TestProjektic.
```

Ako deinstalacija ne uspije

Onda prvo deinstalirajte međuovisne pakete, a tek onda sam Entity Framework.

Rad kroz GUI

- Instalacija / deinstalacija se može obaviti i kroz GUI, gdje po tome da li se pored paketa vidi zelena strelica ili gumb 'Install' znamo da li je EF instaliran ili ne:



Kreiranje modela iz baze (I)

1. Desni klik na projekt

2.

3.

4.

5.

6. Upišite ime modela

7.

8.

9.

Entity Data Model Wizard

Choose Model Contents

What should the model contain?

EF Designer from database

Empty EF Designer model

Empty Code First model

Code First from database

Creates a model in the EF Designer based on an existing database. You can choose the database connection, settings for the model, and database objects to include in the model. The classes your application will interact with are generated from the model.

< Previous Next > Finish Cancel

Add New Item - WebApplication1

Installed

Visual C#

Code

Data

General

Web

General

Markup

MVC

Razor

Scripts

SignalR

Web API

Web Forms

SideWaffle

Windows Forms

WPF

AppVeyor

Azure

Build and NuGet

Extensibility

LightSwitch

Misc

PowerShell

Reporting

Silverlight

SQL Server

ADO.NET Entity Data Model

Visual C#

DataSet

Visual C#

EF 5.x DbContext Generator

Visual C#

EF 6.x DbContext Generator

Visual C#

LINQ to SQL Classes

Visual C#

SQL Server Database

Visual C#

XML File

Visual C#

XML Schema

Visual C#

XSLT File

Visual C#

Type: Visual C#

A project item for creating an ADO.NET Entity Data Model.

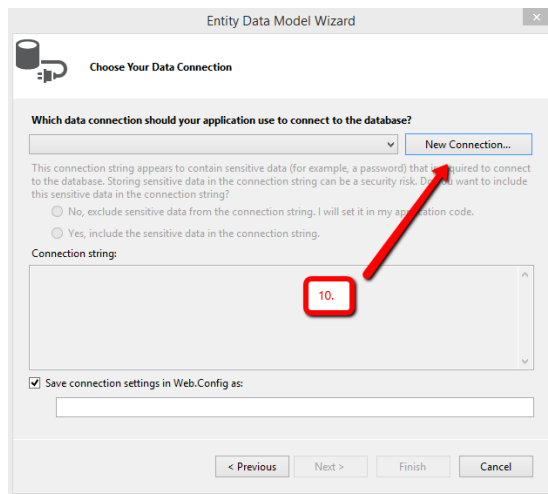
Click here to go online and find templates.

Name: Konferencija

Add Cancel

Kreiranje modela iz baze (II)

VISOKO UČILIŠTE



Entity Data Model 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.

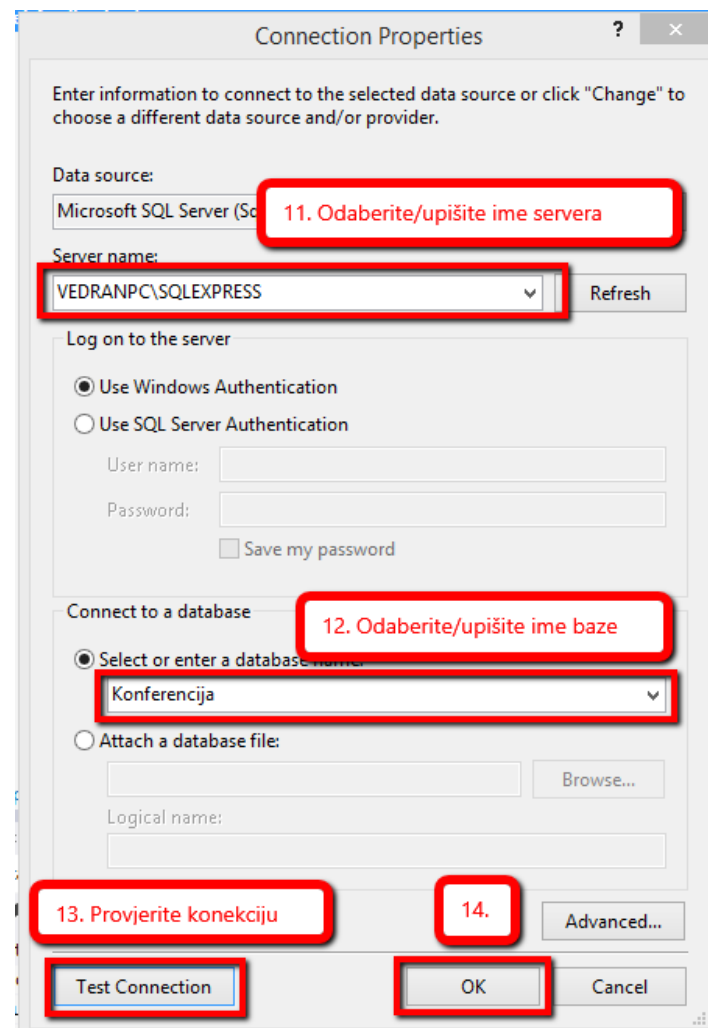
☐ Yes, include the sensitive data in the connection string.

Connection string:

10.

☒ Save connection settings in Web.Config as:

< Previous Next > Finish Cancel



Connection Properties

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 (S

11. Odaberite/upišite ime servera

Server name:

VEDRANPC\SQLEXPRESS Refresh

Log on to the server

☒ Use Windows Authentication

☐ Use SQL Server Authentication

User name:

Password:

☐ Save my password

Connect to a database

12. Odaberite/upišite ime baze

☒ Select or enter a database name:

Konferencija

☐ Attach a database file:

Browse...

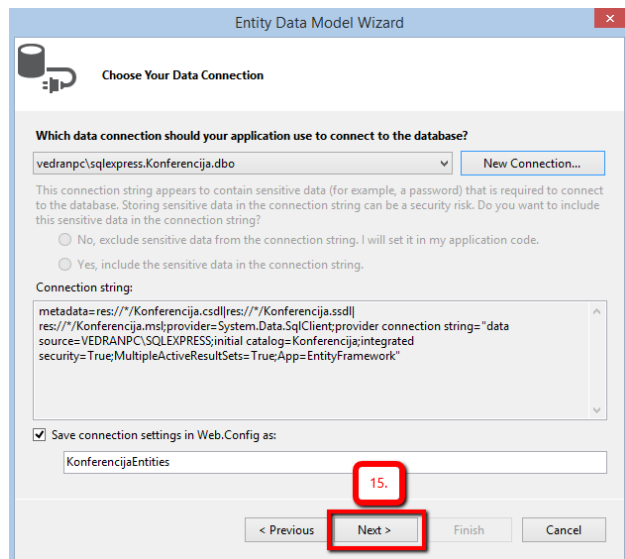
Logical name:

13. Provjerite konekciju

14.

Advanced...

Test Connection OK Cancel



Entity Data Model Wizard

Choose Your Data Connection

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

vedranpc\sqlexpress.Konferencija.dbo 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:

metadata=res://*/Konferencija.csdl|res://*/Konferencija.ssdl|res://*/Konferencija.msl|provider=System.Data.SqlClient;provider connection string="data source=VEDRANPC\SQLEXPRESS;initial catalog=Konferencija;integrated security=True;MultipleActiveResultSets=True;App=EntityFramework"

☒ Save connection settings in Web.Config as:

KonferencijaEntities 15.

< Previous Next > Finish Cancel

Kreiranje modela iz baze (III)

Entity Data Model Wizard

Choose Your Database Objects and Settings

Which database objects do you want to include in your model?

☒ Tables

- ☒ dbo
 - ☒ History
 - ☒ Hoteli
 - ☒ Hoteli_Backup
 - ☒ Polaznici
 - ☒ Povijest
 - ☒ Predavaci
 - ☒ Predavanja
 - ☒ PrijaveNaPredavanja
 - ☒ Racuni
 - ☒ sDvorane
 - ☒ sGradovi
 - ☒ sSmjerovi
 - ☒ sTipoviPredavanja
 - ☒ sTvrteke
 - ☐ sysdiagrams
 - ☐ T
 - ☐ Tsimple
- ☒ test
- ☐ Views
- ☐ Stored Procedures and Functions

☒ Pluralize or singularize generated object names

☒ Include foreign key columns in the model

☒ Import selected stored procedures and functions into the entity model

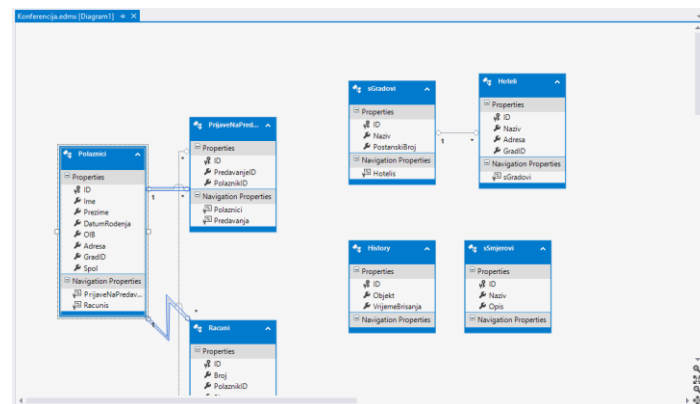
Model Namespace:
KonferencijaModel

16. Odaberite željene tablice, poglede (Views), spremljene procedure ili funkcije

17. Označite sve opcije. Ne morate mijenjati ime Model Namespace-a, osim ako nemate razlog za to.

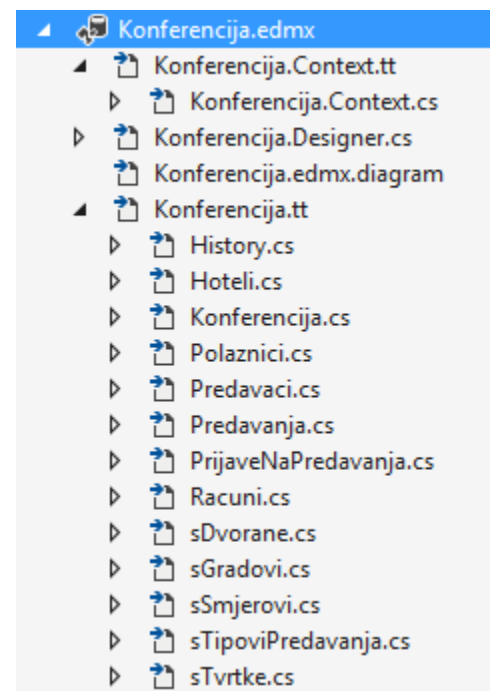
18.

< Previous Next > Finish Cancel



Što nam donosi model?

- Nakon što klikom na 'Save' spremimo promjene modela, generiraju se C# klase koje možemo koristiti za manipuliranje objektima i podacima unutar našeg projekta



- Entiteti EF-a su MODELI našeg MVC projekta

Validacija podataka modela

- MVC model kreiran od strane Entity Framework komponente:
 - Rukuje automatski greškama vezanim na tipove podataka
 - Regenerira se svaki put kada se projekt kompajlira.
 - Prilagođena logika će biti izgubljena zbog regeneriranja koda

... Ili drugim riječima imamo ...

- **PROBLEM:** primjenimo li validacijske attribute direktno na svojstva entiteta unutar generiranih klasa, slijedeći put kada modificiramo EF model i spremimo ga – **izgubiti ćemo sve što smo sami nadopisali** – jer će se sve klase ponovo regenerirati!

Kako onda proširiti model/entitet?

- Kako proširiti entitete?
- Koristimo: Partial Class
- Kreiramo odvojenu meta klasu u svrhu definiranja grešaka vezanih na poslovna pravila.

```
public partial class Blog
{
}
```

```
public class BlogMetadata
{
    Required(ErrorMessage="Naziv je obavezan")]
    public object Naziv { get; set; }

    Required(ErrorMessage="Opis je obavezan")]
    public object Opis { get; set; }
}
```

Primjer MetadataType klase

```
using System;
using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;
using System.Runtime.Serialization;

namespace DAL
{
    [DataContract]
    [MetadataType(typeof(TfsServerMetadata))]
    [Serializable]
    31 references | 0 authors | 0 changes
    public partial class TfsServer
    {
    }

    1 reference | 0 authors | 0 changes
    public partial class TfsServerMetadata
    {
        [Editable(false)]
        0 references | 0 authors | 0 changes
        public int Id { get; set; }

        [Required(ErrorMessage = "Name is required.")]
        [Display(Name = "Name")]
        0 references | 0 authors | 0 changes
        public string Name { get; set; }

        [Required(ErrorMessage = "Description is required.")]
        [Display(Name = "Description")]
        0 references | 0 authors | 0 changes
        public string Description { get; set; }
    }
}
```

