1-How to install Microsoft SQL Server 2012 Express LocalDB with the SqlLocalDB.MSI on your desktop or laptop?

* Download and install it
* Go to the command prompt and type: SqlLocalDB info
* In order to create an automatic LocalDB instance, go to the command prompt and type: SqlLocalDB create v11.0

2- How to install Entity Framework in Visual Studio?

* In the package manager console, type: Install-Package EntityFramework

3-How to create the CRUD actions methods and views?

* I first created the model “Movie.cs” with the corresponding attributes and methods
* I then created the connection string in the web.config
* I then created the controller “MoviesController.cs” by selecting “MVC5 Controller with Views using Entity Framework”.

Therefore, the folder “Movies” containing Create.cshtml, Delete.cshtml, Details.cshtml, and Index.cshtml will be created automatically by the Entity Framework.

In a brief detail:

* Adding Model Classes:
* public class Movie
* {
* public int ID { get; set; }
* [Required]
* public string Title{get;set;}
* [DataType(DataType.Date)]
* public DateTime ReleaseDate{get;set;}
* [Required]
* public string Genre{get;set;}
* [Range(1, 100)]
* [DataType(DataType.Currency)]
* public decimal Price{get;set;}
* [StringLength(5)]
* public string Rating{get;set;}
* }
* public class MovieDBContext:DbContext
* {
* public DbSet<Movie> Movies { get; set; }
* }
* Creating a Connection String and Working with SQL Server LocalDB: Open the application root Web.config file. (Not the Web.config file in the Views folder.) Open the Web.config file outlined in red.
* Add <connectionStrings> <add name="DefaultConnection" connectionString="Data Source=(LocalDb)\v11.0;Initial Catalog=aspnet-MvcMovie-2012213181139;Integrated Security=true" providerName="System.Data.SqlClient" /> <add name="MovieDBContext" connectionString="Data Source=(LocalDB)\v11.0;AttachDbFilename=|DataDirectory|\Movies.mdf;Integrated Security=True" providerName="System.Data.SqlClient" /> </connectionStrings>
* Next, you'll build a new MoviesController class that you can use to display the movie data and allow users to create new movie listings
* Managing Model and Database Schema Differences
* Automatically Re-Creating the Database on Model Changes:

1- right click the Models folder, select Add, and then select New Item.

2- In the Add New Item dialog, select Class then name the class "MovieInitializer". Update the MovieInitializer class to contain the following code:

using System;

using System.Collections.Generic;

using System.Data.Entity;

namespace MvcMovie.Models

{

public class MovieInitializer : DropCreateDatabaseIfModelChanges<MovieDBContext>

{

protected override void Seed(MovieDBContext context)

{

var movies = new List<Movie>{

new Movie{

Title = "When Harry Met Sally",

ReleaseDate = DateTime.Parse("1989-1-11"),

Genre = "Romantic Comedy",

Rating = "R",

Price = 7.99M},

new Movie{

Title = "Ghostbusters ",

ReleaseDate = DateTime.Parse("1984-3-13"),

Genre = "Comedy",

Rating = "R",

Price = 8.99M},

new Movie{

Title = "Ghostbusters 2",

ReleaseDate = DateTime.Parse("1986-2-23"),

Genre = "Comedy",

Rating = "R",

Price = 9.99M},

new Movie{

Title = "Rio Bravo",

ReleaseDate = DateTime.Parse("1959-4-15"),

Genre = "Western",

Rating = "R",

Price = 3.99M}

};

movies.ForEach(d => context.Movies.Add(d));

}

}

}

3- Open the Global.asax filet:

At the beginning of the Application\_Start method, add a call to Database.SetInitializer as shown below:

using MvcMovie.Models; // MovieInitializer

using System;

using System.Collections.Generic;

using System.Data.Entity;

using System.Data.Entity.Infrastructure;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using System.Web.Optimization;

using System.Web.Routing;

namespace MvcMovie

{

public class MvcApplication : System.Web.HttpApplication

{

protected void Application\_Start()

{

//added by eddddyyyyyyyyyyyyyyyyyyyyyyyy

Database.SetInitializer<MovieDBContext>(new MovieInitializer());

AreaRegistration.RegisterAllAreas();

// Use LocalDB for Entity Framework by default eddddddyyyyyyyyyy

Database.DefaultConnectionFactory = new SqlConnectionFactory("Data Source = (localdb)\v11.0; Integrated Security = True; MultipleActiveResultSets = True");

FilterConfig.RegisterGlobalFilters(GlobalFilters.Filters);

RouteConfig.RegisterRoutes(RouteTable.Routes);

BundleConfig.RegisterBundles(BundleTable.Bundles);

}

}

}

Put the cursor on the red squiggly line (on MovieDBContext, orMovieInitializer ) right click and select

Resolve, then using MvcMovie.Models;