

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

ایجاد کردن یک connection string و کار با نسخه ی SQL Server LocalDB

مدرس : مهندس افشین رفوآ

دوره آموزش MVC

ایجاد کردن یک connection string و کار با نسخه ی SQL Server LocalDB

کلاس **MovieDbContext** که ایجاد کردید، وظیفه ی اتصال به پایگاه داده و نگاشت اشیا **movie** به سطر (رکورد) های پایگاه داده را بر عهده دارد. یکی از سوال هایی که ممکن است برای شما پیش آید این است که چگونه می توان تعیین کرد به کدام پایگاه داده متصل شود. در پاسخ باید گفت که نیازی نیست مشخص کنید کدام پایگاه داده مورد استفاده قرار گیرد، **EF** خود به صورت پیش فرض به پایگاه داده ی **LocalDB** متصل می شود. در این بخش به صورت صریح یک **connection string** در فایل **Web.config** برنامه خود اضافه می کنیم.

SQL Server Express LocalDB

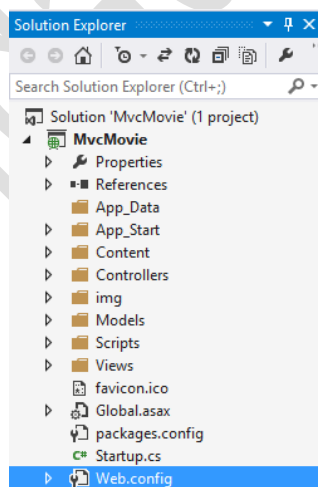
**LocalDB** یک نسخه ی سبک وزن و کم حجم از موتور پایگاه داده ی **SQL Server Express** است که به مجرد تقاضا راه اندازی شده و در حالت **user mode** اجرا می شود. **LocalDB** در یک حالت اجرایی ویژه ی **SQL Server Express** اجرا می شود که به شما اجازه می دهد با پایگاه داده به صورت فایل های **.mdf** کار کنید. به طور معمول، فایل های پایگاه داده ی **LocalDB** در پوشه ی **App\_Data** پروژه ی وب نگه داری می شود.

SQL Server Express به هیچ وجه برای استفاده در برنامه های تحت وب تولیدی ( production web applications ) توصیه نمی شود. این امر به خصوص در مورد LocalDB صدق می کند؛ به هیچ وجه نباید LocalDB برای تولید با یک برنامه ی تحت وب بکار رود زیرا این نسخه اساسا برای کار با IIS (سرویس های اطلاعات اینترنتی) طراحی نشده است. با این حال، یک پایگاه داده ی LocalDB را می توان به راحتی به SQL Server یا SQL Azure انتقال (migrate) داد.

در ویرایش های 2012 و 2013 محیط برنامه نویسی Visual Studio ، LocalDB به صورت پیش فرض همراه با خود Visual Studio نصب می شود.

به صورت پیش فرض، Entity Framework به دنبال یک connection string که دارای اسمی یکسان با کلاس object context (برای این پروژه ی خاص MovieDbContext) می باشد، می گردد.

فایل Web.config که در زیر به نمایش گذاشته شده را باز کنید. (منظور آن فایل Web.config که در پوشه ی Views قرار دارد نیست.)



المان <connectionStrings> را پیدا کنید:

```

<?xml version="1.0" encoding="utf-8"?>
<!--
For more information on how to configure your ASP.NET application, please visit
http://go.microsoft.com/fwlink/?LinkId=301880
-->
<configuration>
  <configSections>
    <!-- For more information on Entity Framework configuration, visit http://go
    <section name="entityFramework" type="System.Data.Entity.Internal.ConfigFile
  </configSections>
  <connectionStrings>
    <add name="DefaultConnection" connectionString="Data Source=(LocalDb)\v11.0;
    providerName="System.Data.SqlClient" />
  </connectionStrings>
  <appSettings>
    <add key="webpages:Version" value="3.0.0.0" />
    <add key="webpages:Enabled" value="false" />
    <add key="PreserveLoginUrl" value="true" />
    <add key="ClientValidationEnabled" value="true" />
    <add key="UnobtrusiveJavaScriptEnabled" value="true" />
  </appSettings>
</system.web>

```

Connection string زیر را به المان `<connectionStrings>` داخل `Web.config` اضافه نمایید:

```

<add name="MovieDBContext"
connectionString="Data Source=(LocalDB)\v11.0;AttachDbFilename=|DataDirectory|\Movies.mdf;Integrated
Security=True"
providerName="System.Data.SqlClient"
/>

```

مثال زیر بخشی از فایل `Web.config` را نشان می دهد که `connection string` جدید به آن اضافه شده است:

```

<connectionStrings>
  <add name="DefaultConnection" connectionString="Data
Source=(LocalDb)\v11.0;AttachDbFilename=|DataDirectory|\aspnet-MvcMovie-20130603030321.mdf;Initial
Catalog=aspnet-MvcMovie-20130603030321;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>

```

دو `connection string` بسیار شبیه هستند. اولین `connection string`، `DefaultConnection` نامیده شده و برای مدیریت عضویت در پایگاه داده و اینکه چه کسی می تواند به برنامه (`membership database`) دسترسی داشته باشد، بکار می رود. `Connection string` ای که شما اضافه کرده اید یک پایگاه داده `LocalDB` به نام `Movie.mdf` تعریف می کند که در پوشه `App_Data` قرار می گیرد.

اسم **connection string** باید با اسم کلاس **DbContext** مطابقت داشته باشد.

```
using System;
using System.Data.Entity;
namespace MvcMovie.Models
{
    public class Movie
    {
        public int ID { get; set; }
        public string Title { get; set; }
        public DateTime ReleaseDate { get; set; }
        public string Genre { get; set; }
        public decimal Price { get; set; }
    }
    public class MovieDbContext : DbContext
    {
        public DbSet<Movie> Movies { get; set; }
    }
}
```

در واقع نیازی به افزودن **connection string** : **MovieDbContext** نیست. در صورت مشخص نکردن **connection string**، **EF** خود یک پایگاه داده ی **LocalDB** به همراه اسم کامل کلاس **DbContext** (در این مورد **MvcMovie.Models.MovieDbContext**) در پوشه ی **users** ایجاد می کند. می توانید اسم پایگاه داده را مطابق میل خود انتخاب کنید مادام اینکه دارای پسوند **MDF** باشد. به عنوان مثال می توان اسم پایگاه داده ی مورد نظر را **MyFilms.mdf** انتخاب کرد.