ASP.NET

Home   Get Started   Learn   Hosting   Downloads   Community   Forums   Help

Adding a Model

By Rick Anderson | last updated May 28, 2015
1063 of 1131 people found this helpful

In this section you'll add some classes for managing movies in a database. These classes will be the "model" part of the ASP.NET MVC app.

You'll use a .NET Framework data-access technology known as the **Entity Framework (http://msdn.microsoft.com/en-us/library/bb399572(VS.110).aspx)** to define and work with these model classes. The Entity Framework (often referred to as EF) supports a development paradigm called *Code First*. Code First allows you to create model objects by writing simple classes. (These are also known as POCO classes, from "plain-old CLR objects.") You can then have the database created on the fly from your classes, which enables a very clean and rapid development workflow. If you are required to create the database first, you can still follow this tutorial to learn about MVC and EF app development. You can then follow Tom Fizmakens **ASP.NET Scaffolding (/visual-studio/overview/2013/aspnet-scaffolding-overview)** tutorial, which covers the database first approach.

Adding Model Classes

In **Solution Explorer**, right click the *Models* folder, select **Add**, and then select **Class**.
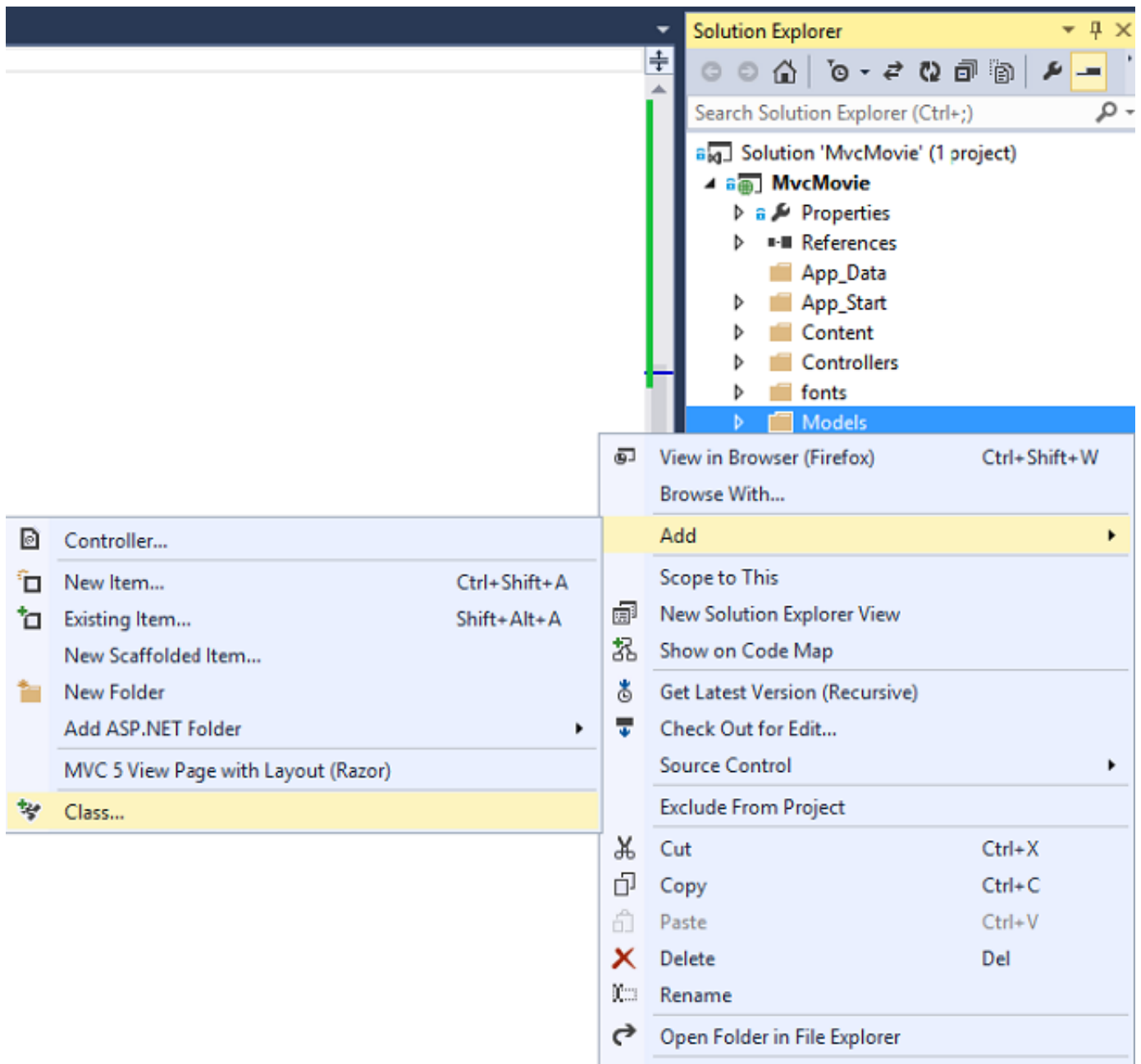
Enter the *class* name "Movie".

Add the following five properties to the `Movie` class:

```
using System;

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; }
    }
}
```

We'll use the `Movie` class to represent movies in a database. Each instance of a `Movie` object will correspond to a row within a database table, and each property of the `Movie` class will map to a column in the table.

In the same file, add the following `MovieDBContext` class:

```
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; }
    }
}
```

The `MovieDBContext` class represents the Entity Framework movie database context, which handles fetching, storing, and updating `Movie` class instances in a database. The `MovieDBContext` derives from the **DbContext (http://msdn.microsoft.com/en-us/library/system.data.entity.dbcontext(v=VS.103).aspx)** base class provided by the Entity Framework.

In order to be able to reference `DbContext` and `DbSet`, you need to add the following `using` statement at the top of the file:

```
using System.Data.Entity;
```

You can do this by manually adding the using statement, or you can right click on the red squiggly lines and click **Resolve**, and then click **using System.Data.Entity.**

Note: Several unused `using` statements have been removed. You can do this by right clicking in the file, click **Organize Usings**, and then click **Remove Unused Usings.**

We've finally added a model (the M in MVC). In the next section you'll work with the database connection string.

This article was originally created on June 10, 2013

Author Information

Rick Anderson – Rick Anderson works as a programmer writer for Microsoft, focusing on ASP.NET MVC, Windows Azure and Entity Framework. You can follow him on twitter via @RickAndMSFT.

Comments (46)