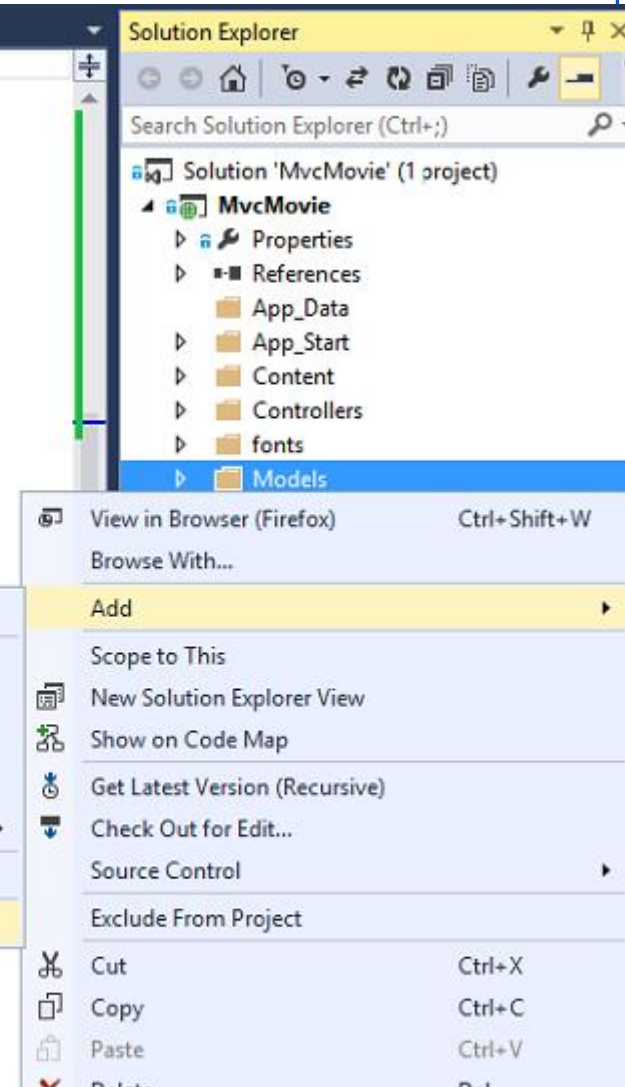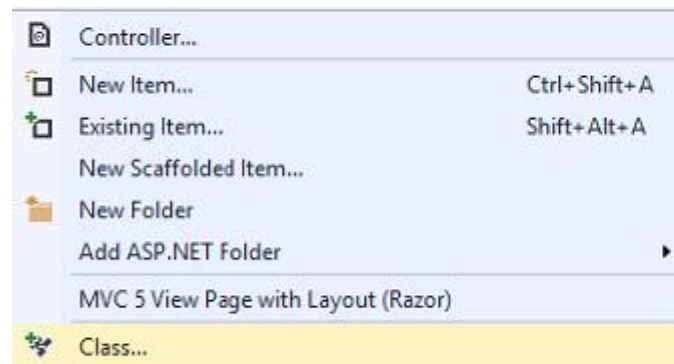# ASP.NET MVC
Scaffolding

Adding Model Classes

- In Solution Explorer, right click the Models folder, select Add, and then select Class

```
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; }
}
```

Add Db

- Add new folder DataAccess
- Add class *<name>*Context

```csharp
public class MovieDBContext : DbContext
{
    public DbSet<Movie> Movies { get; set; }
}

// (The name of the connection string must match the name
// of the  DbContext class)
// Or use the default connectionstring (and perhaps
// rename the file name)

public MovieDBContext()
    : base("DefaultConnection")
{
}
```
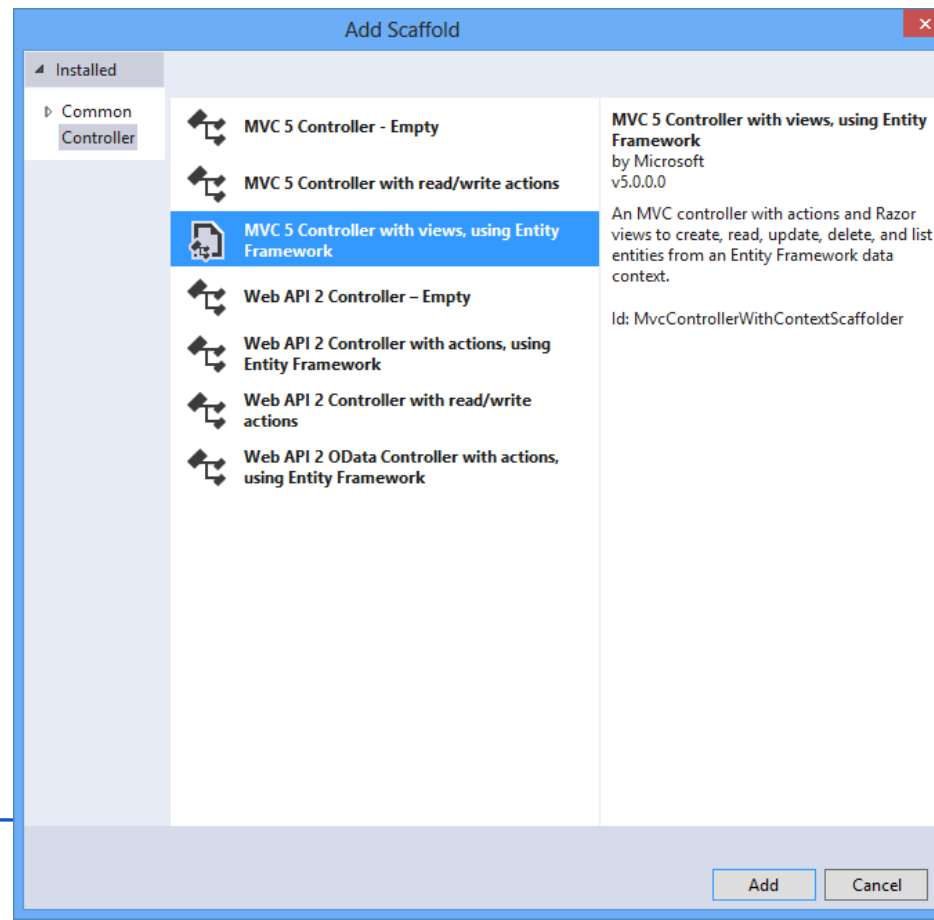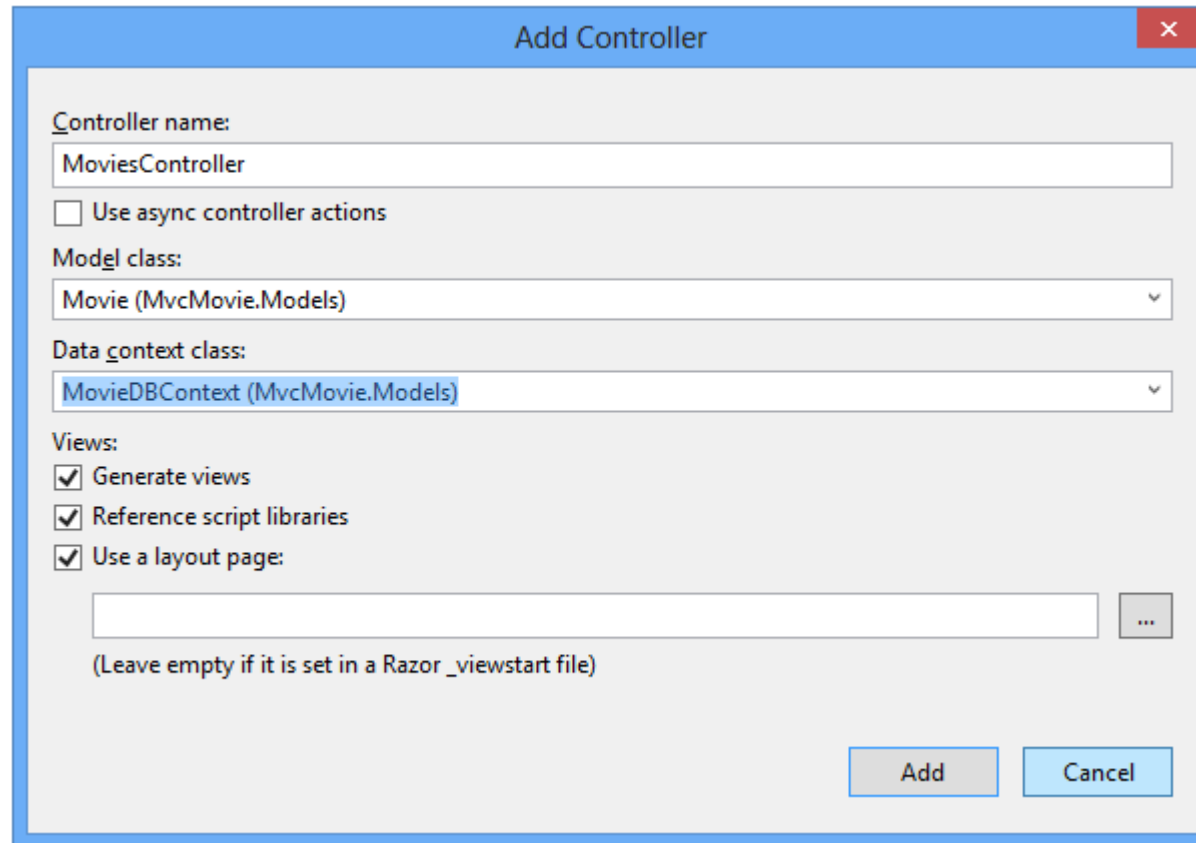
Add Controller

- In Solution Explorer, right-click the *Controllers* folder and then click **Add**, then **Controller**

- In the **Add Scaffold** dialog box, click **MVC 5 Controller with views, using Entity Framework**, and then click **Add**

Remember to build first

Add Controller

- For the Controller name enter **MoviesController**
- Select **Movie (MvcMovie.Models)** for the Model class
- Select **MovieDBContext (MvcMovie.Models)** for the Data context class

Strongly Typed Models

- MVC provides the ability to pass *strongly* typed objects to a view template. This strongly typed approach enables better compile-time checking of your code and richer IntelliSense in the Visual Studio editor.

```
public ActionResult Details(int? id)
{
  if (id == null)
  {
    return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
  }
  Movie movie = db.Movies.Find(id);
  if (movie == null)
  {
    return HttpNotFound();
  }
 return View(movie);
}
```

```
@* In DetailsView *@
@model AspScaffoldingDemo.Models.Movie
```

References & Links

- ASP.NET MVC
  http://asp.net/mvc

AARHUS
UNIVERSITY
SCHOOL OF ENGINEERING