MVC Architectural Pattern

Zouhair Rimale, Ph.D.

Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular

MVC Architectural Pattern

Model

View

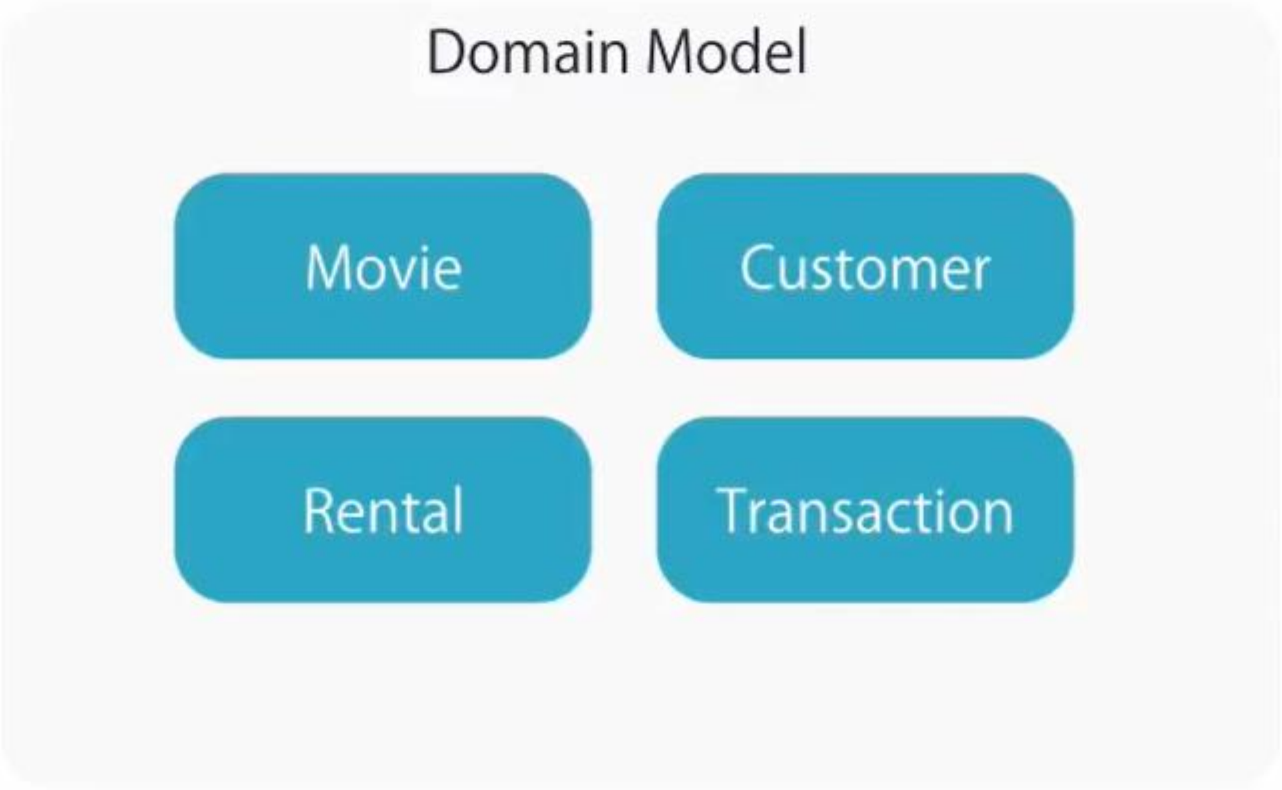
Controller

MVC

- Designed in 1970s
- Widely adopted in web
- ASP.NET MVC, Ruby on Rails, Express

Model

Application data and behaviour in terms of its problem domain, and independent of the UI.



Domain Model

Movie

Customer

Rental

Transaction

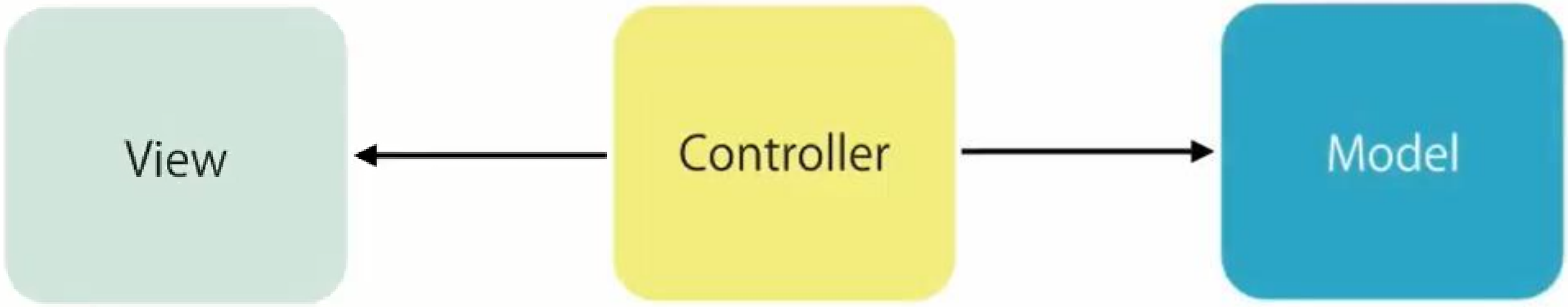
View

The HTML markup that we display to the user.

Controller

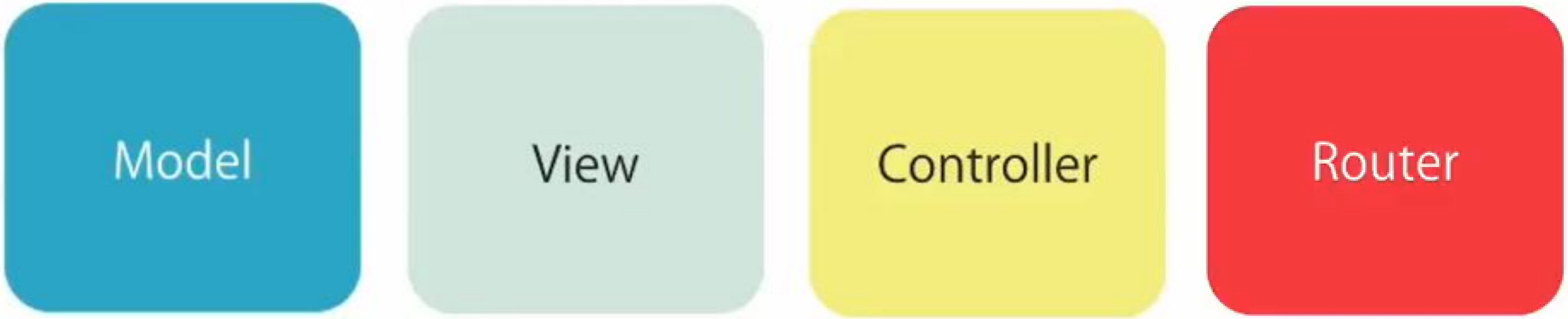
Responsible for handling an HTTP Request.

```
http://vidly.com/movies
```



Better separation of concerns

MVC Architectural Pattern



Model

View

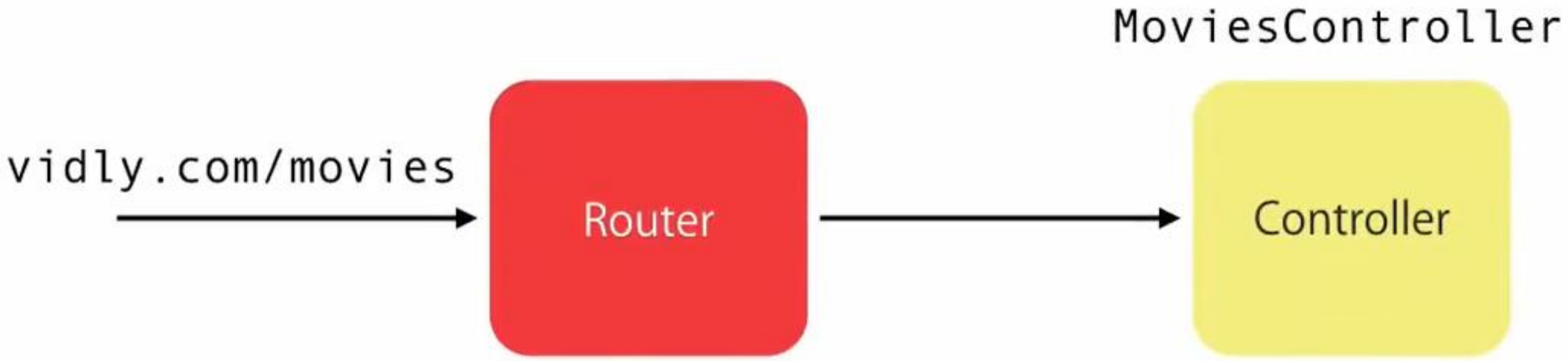
Controller

Router

Router

Selects the right controller to handle a request.

Router



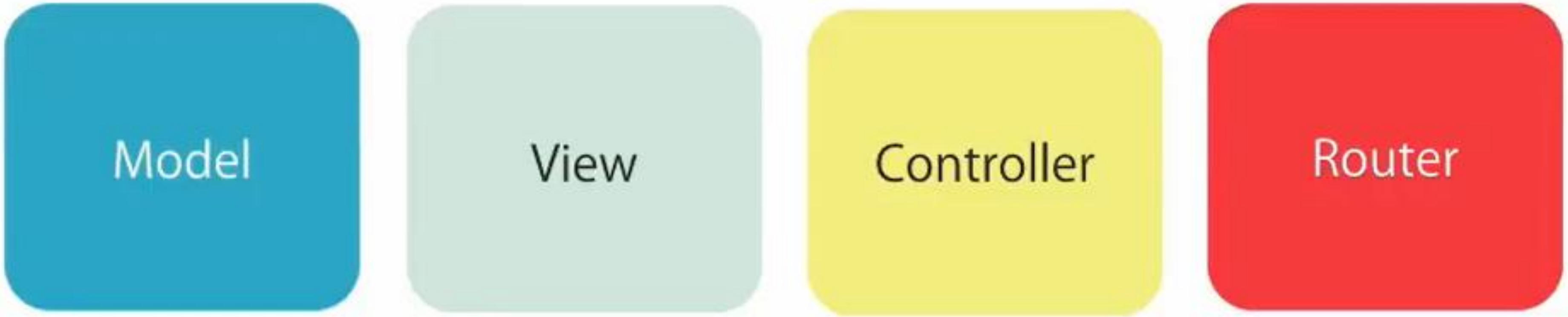
MoviesController

vidly.com/movies

Router

Controller

MVC Architectural Pattern



Model

View

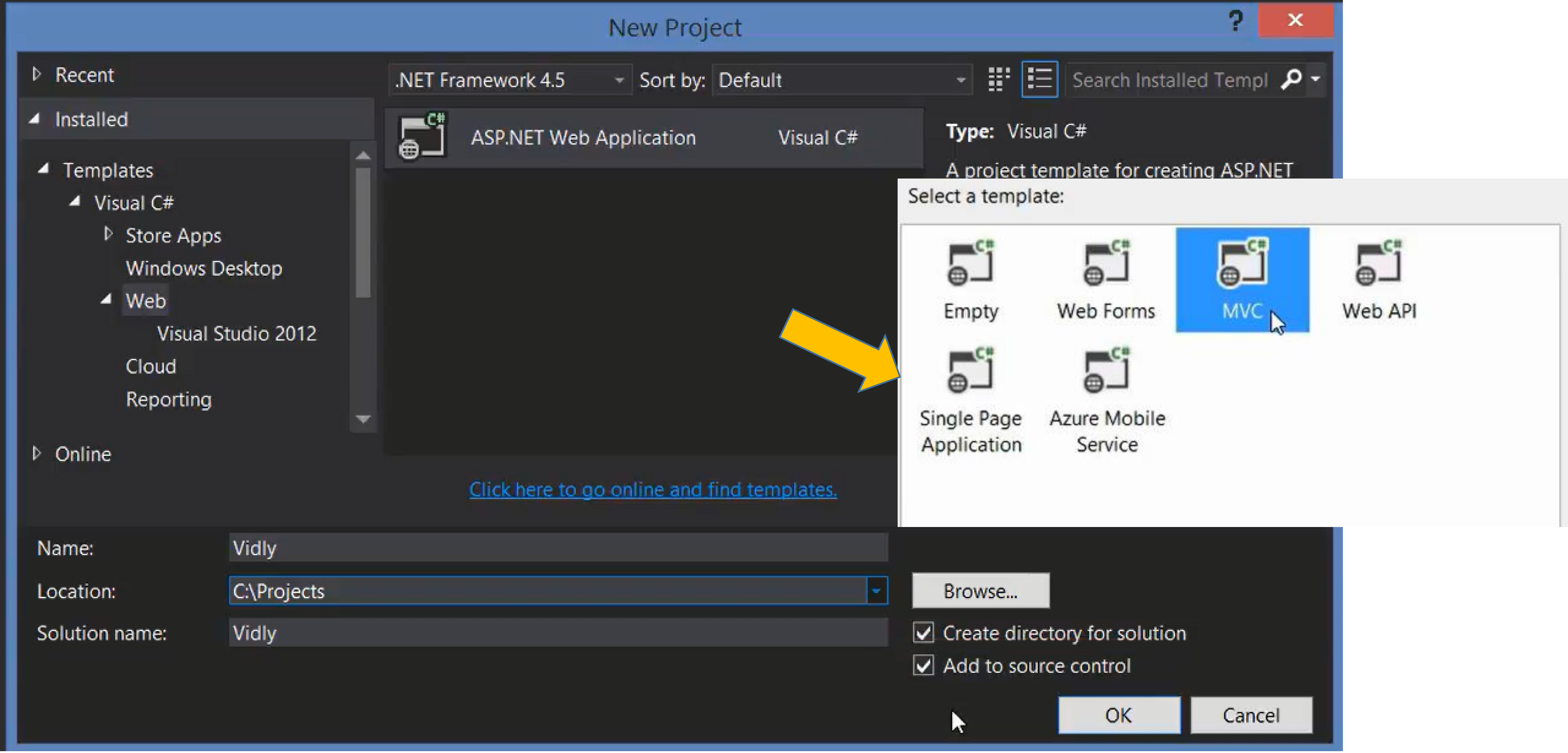
Controller

Router

Demo

Visual Studio 2013 or higher
(Community Edition)

Output



Type: Visual C#

A project template for creating ASP.NET

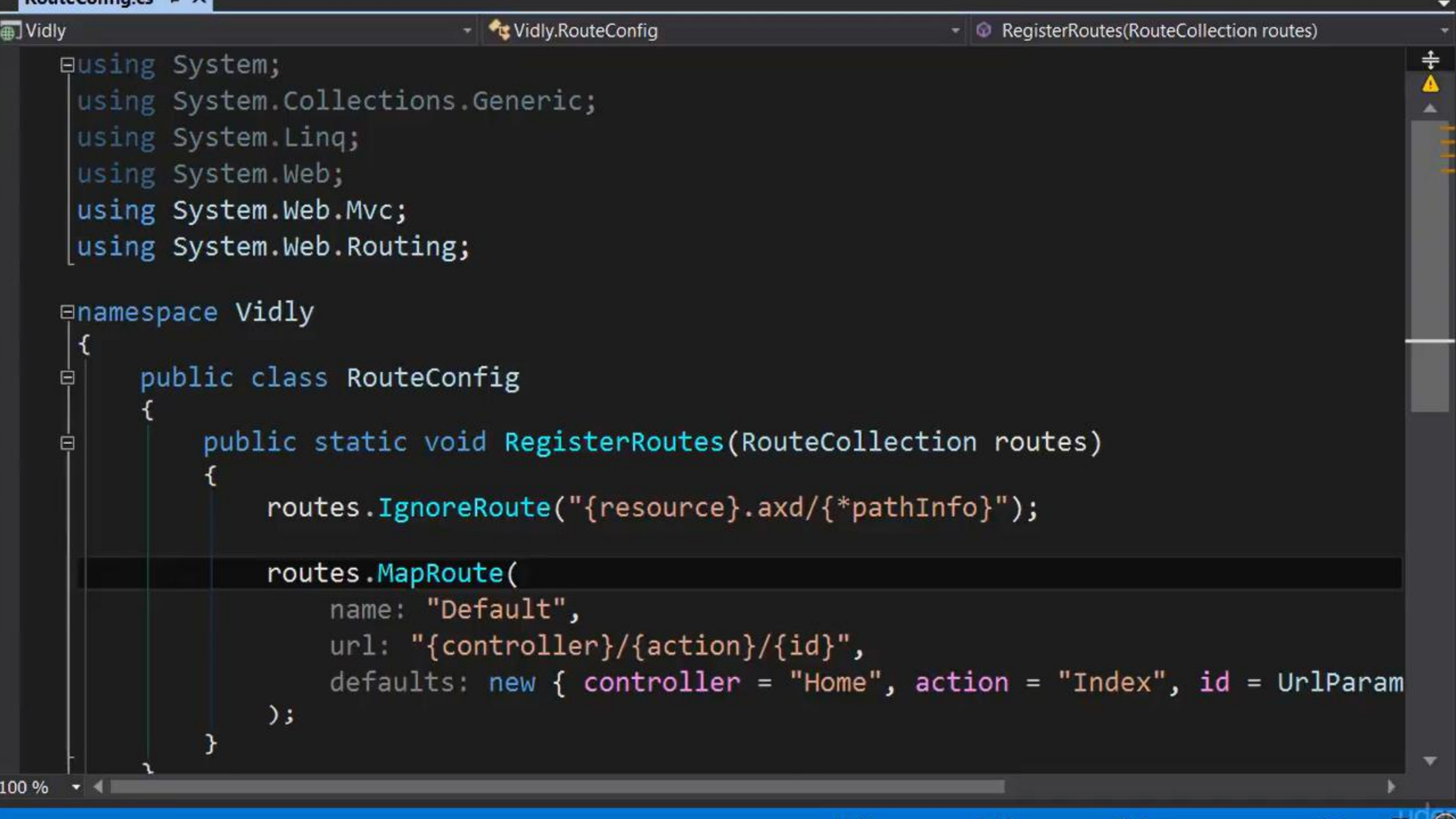
Name: Vidly

Location: C:\Projects

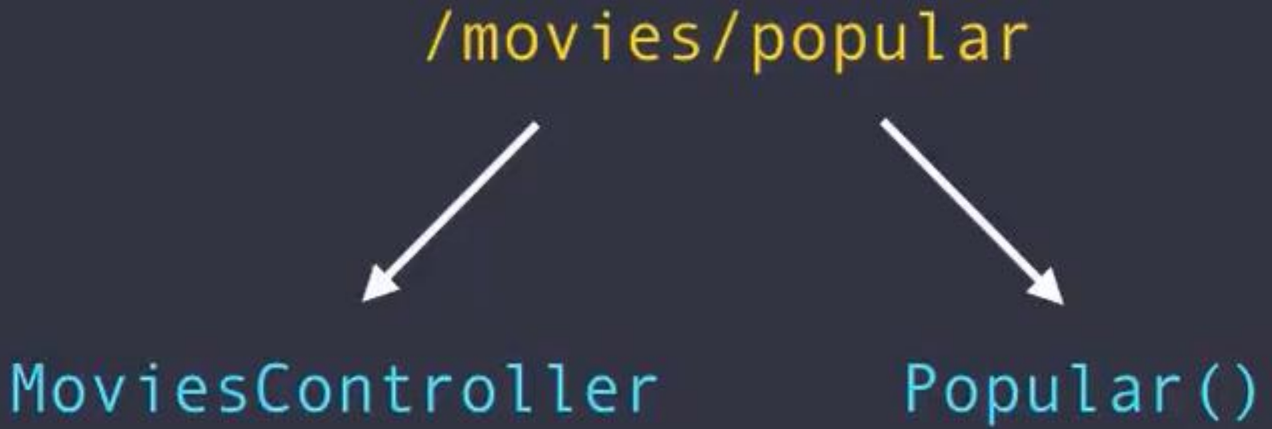
Solution name: Vidly

☑ Create directory for solution

☑ Add to source control



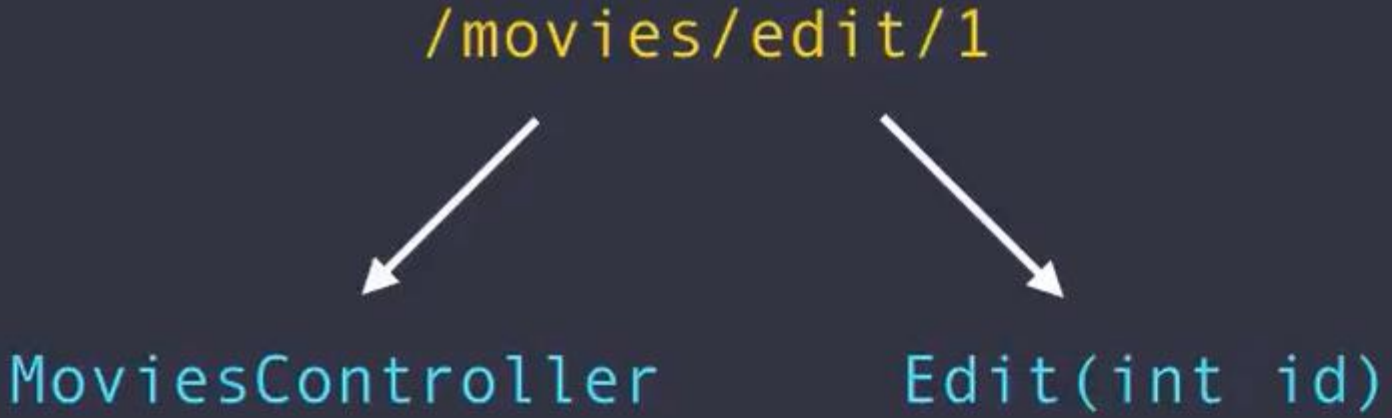
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using System.Web.Routing;

namespace Vidly
{
    public class RouteConfig
    {
        public static void RegisterRoutes(RouteCollection routes)
        {
            routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Home", action = "Index", id = UrlParam
            );
        }
    }
```



/movies/popular

MoviesController

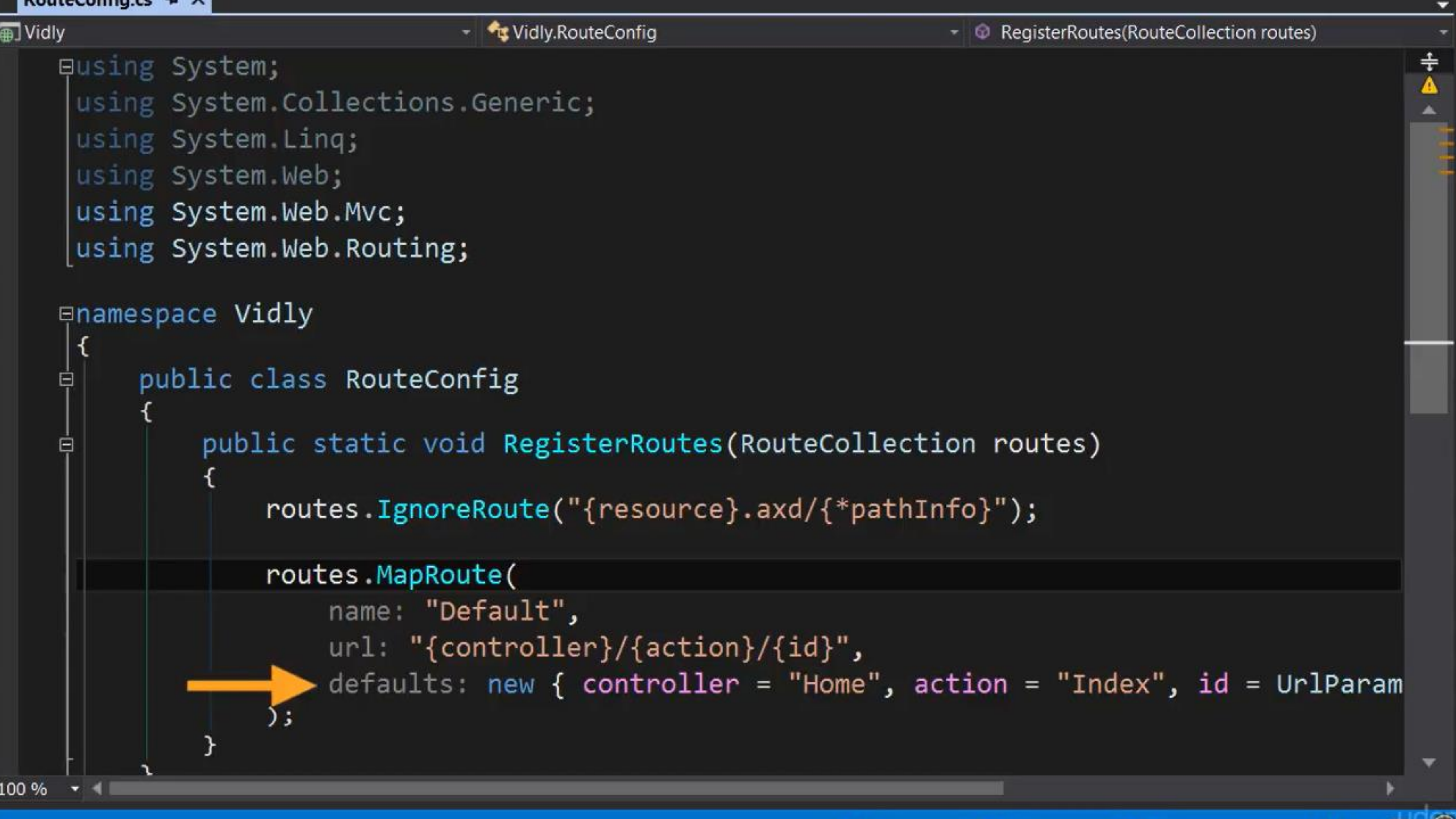
Popular()



/movies/edit/1

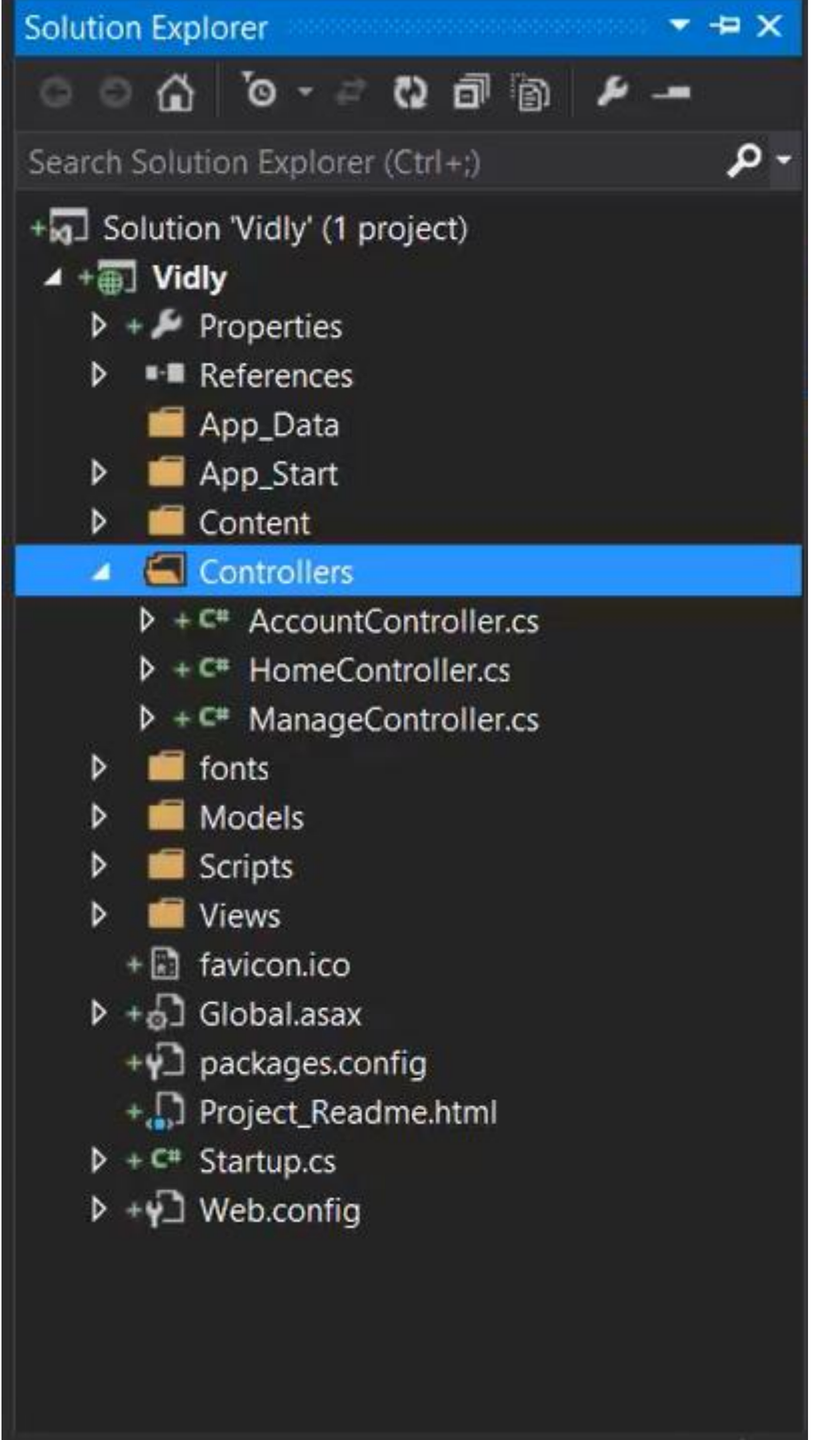
MoviesController

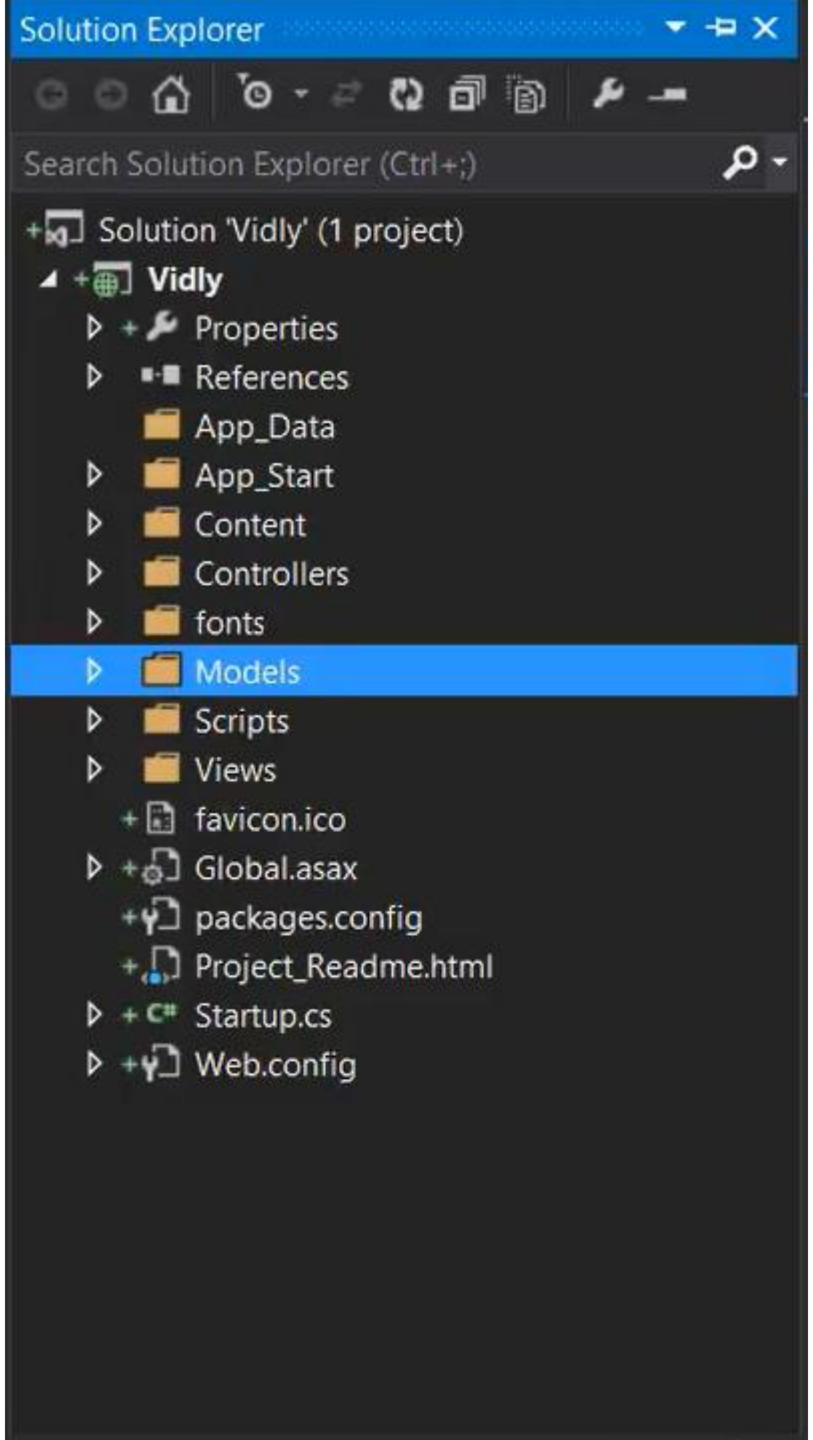
Edit(int id)

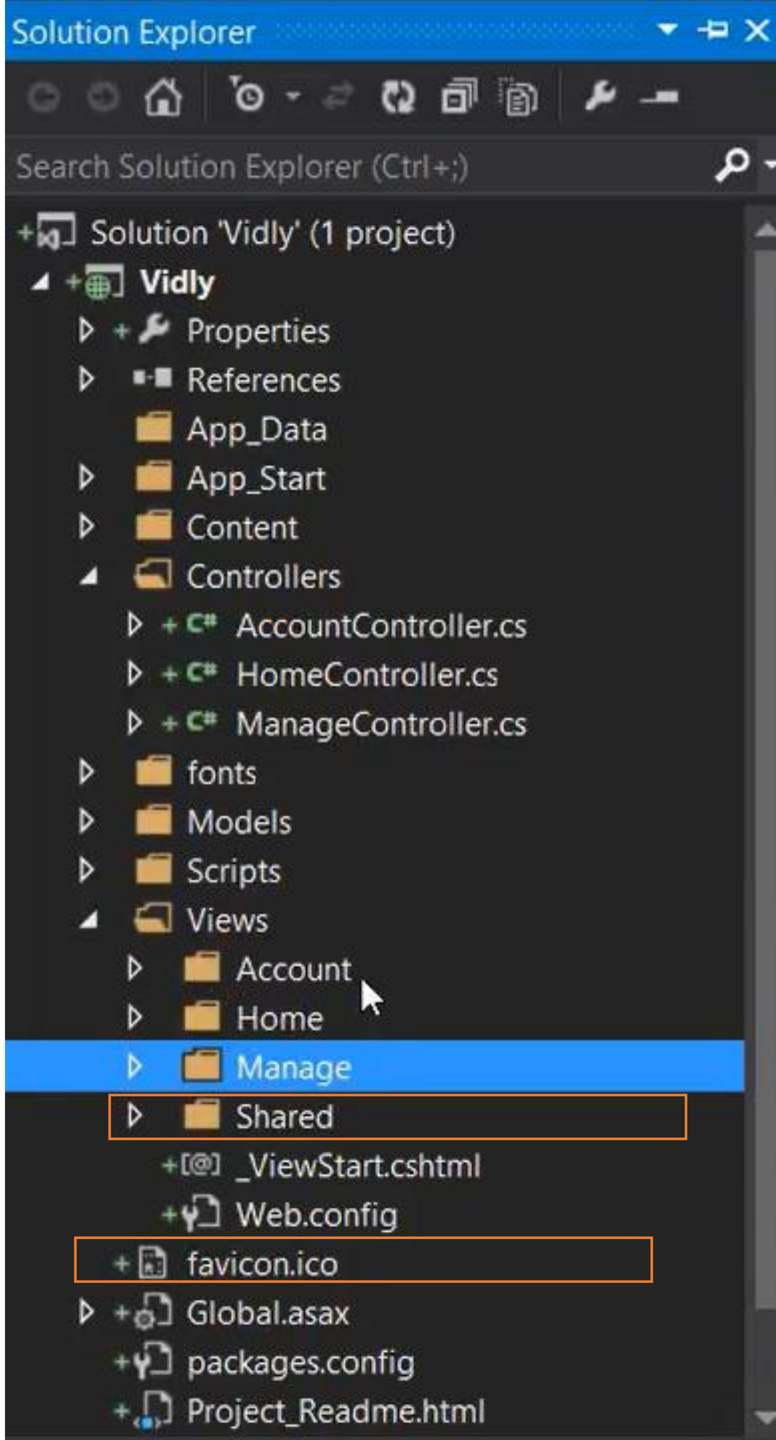


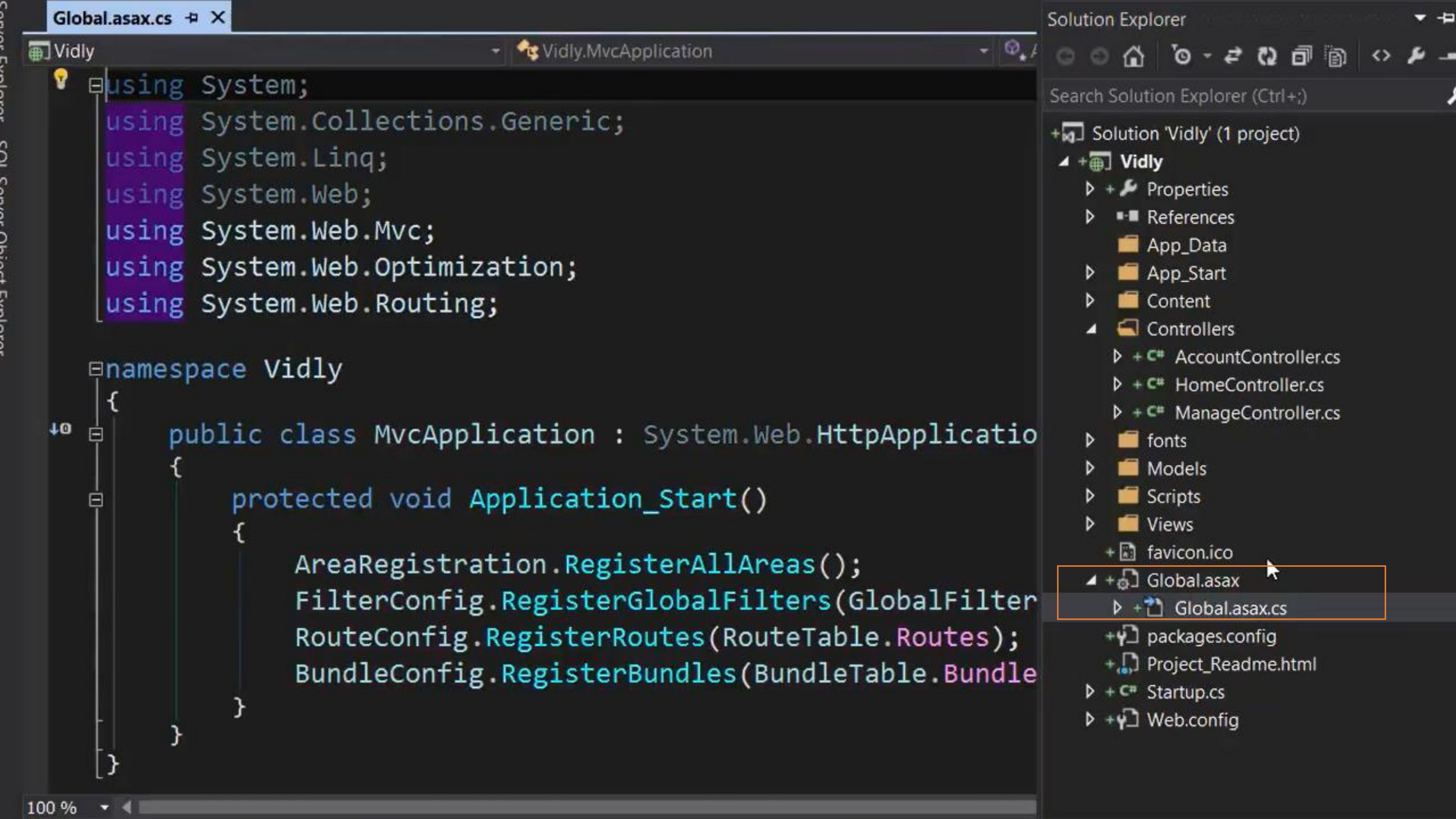
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using System.Web.Routing;

namespace Vidly
{
    public class RouteConfig
    {
        public static void RegisterRoutes(RouteCollection routes)
        {
            routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Home", action = "Index", id = UrlParam
            );
        }
```

```
/movies → MoviesController.Index()
```

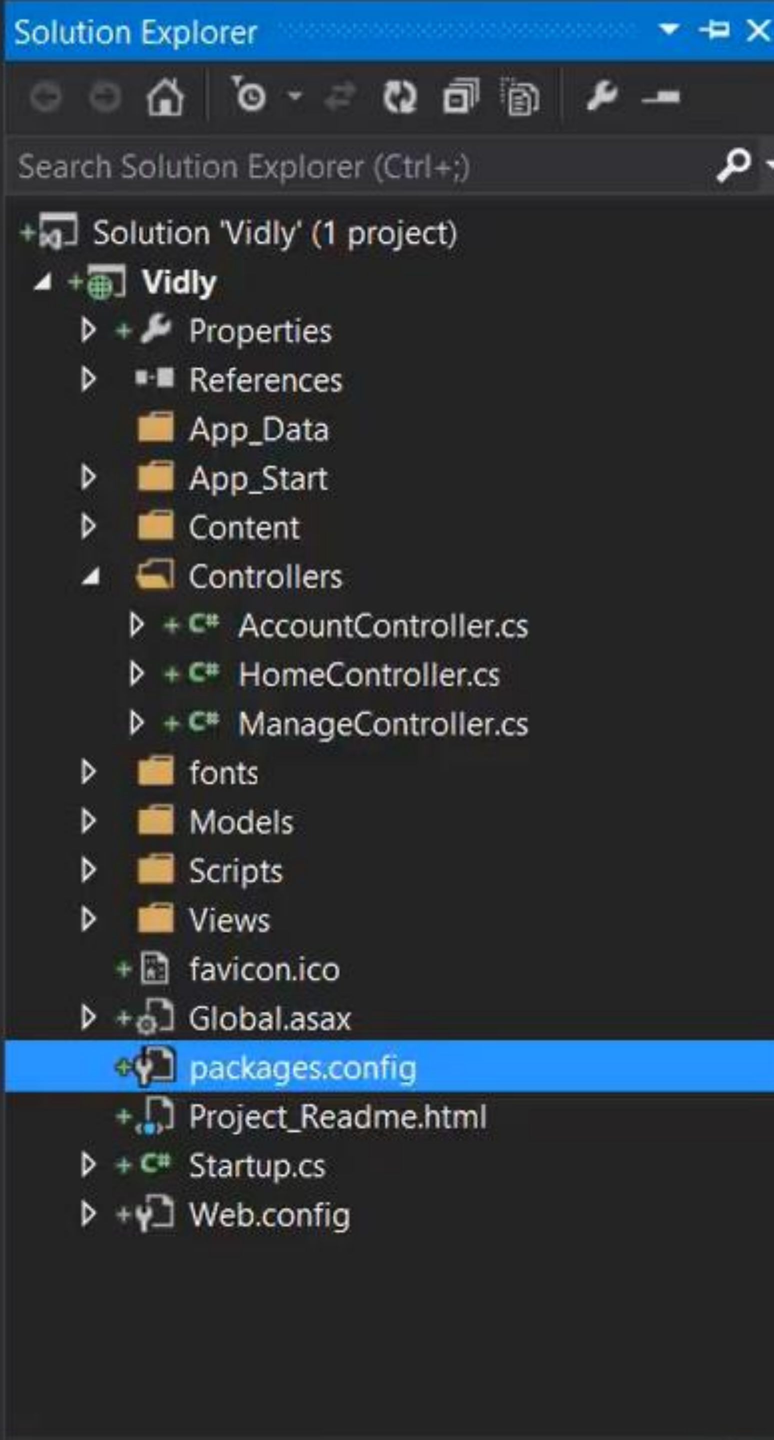

Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'Vidly' (1 project)
Vidly
Properties
References
App_Data
App_Start
Content
Controllers
AccountController.cs
HomeController.cs
ManageController.cs
fonts
Models
Scripts
Views
favicon.ico
Global.asax
packages.config
Project_Readme.html
Startup.cs
Web.config


Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'Vidly' (1 project)
Vidly
Properties
References
App_Data
App_Start
Content
Controllers
fonts
Models
Scripts
Views
favicon.ico
Global.asax
packages.config
Project_Readme.html
Startup.cs
Web.config

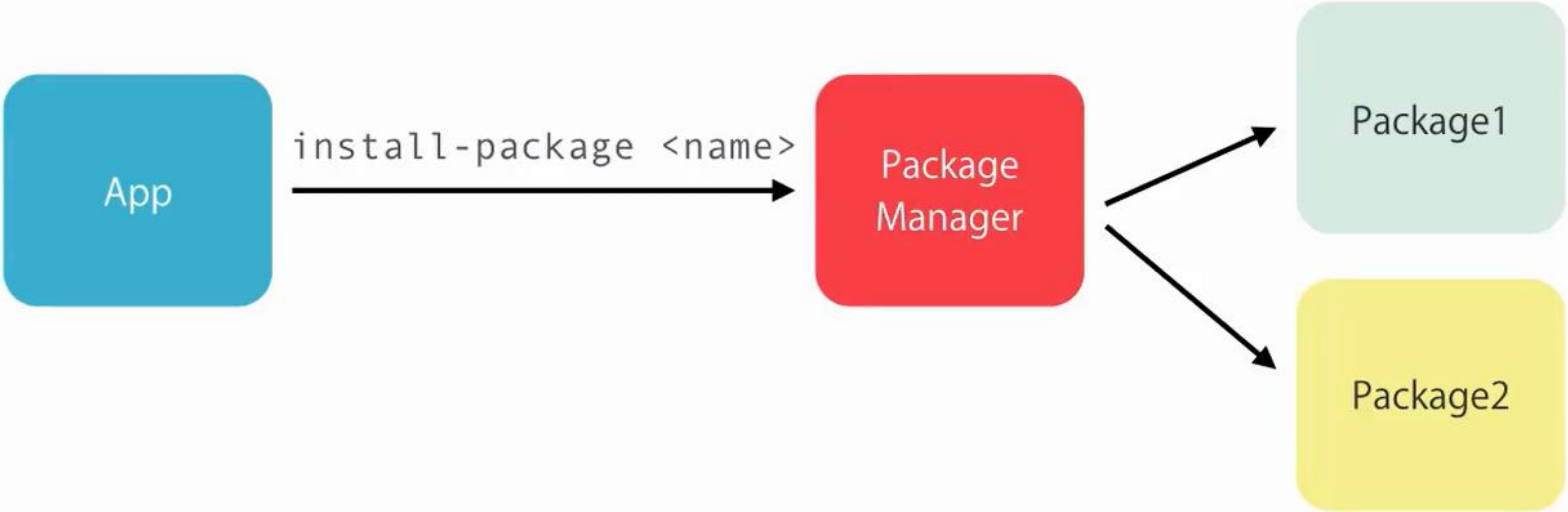

Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'Vidly' (1 project)
Vidly
Properties
References
App_Data
App_Start
Content
Controllers
AccountController.cs
HomeController.cs
ManageController.cs
fonts
Models
Scripts
Views
Account
Home
Manage
Shared
_ViewStart.cshtml
Web.config
favicon.ico
Global.asax
packages.config
Project_Readme.html



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using System.Web.Optimization;
using System.Web.Routing;

namespace Vidly
{
    public class MvcApplication : System.Web.HttpApplicatio
    {
        protected void Application_Start()
        {
            AreaRegistration.RegisterAllAreas();
            FilterConfig.RegisterGlobalFilters(GlobalFilter
            RouteConfig.RegisterRoutes(RouteTable.Routes);
            BundleConfig.RegisterBundles(BundleTable.Bundle
        }
    }
}
```



```
Solution 'Vidly' (1 project)
  Vidly
    Properties
    References
    App_Data
    App_Start
    Content
    Controllers
      AccountController.cs
      HomeController.cs
      ManageController.cs
    fonts
    Models
    Scripts
    Views
    favicon.ico
    Global.asax
    packages.config
    Project_Readme.html
    Startup.cs
    Web.config
```



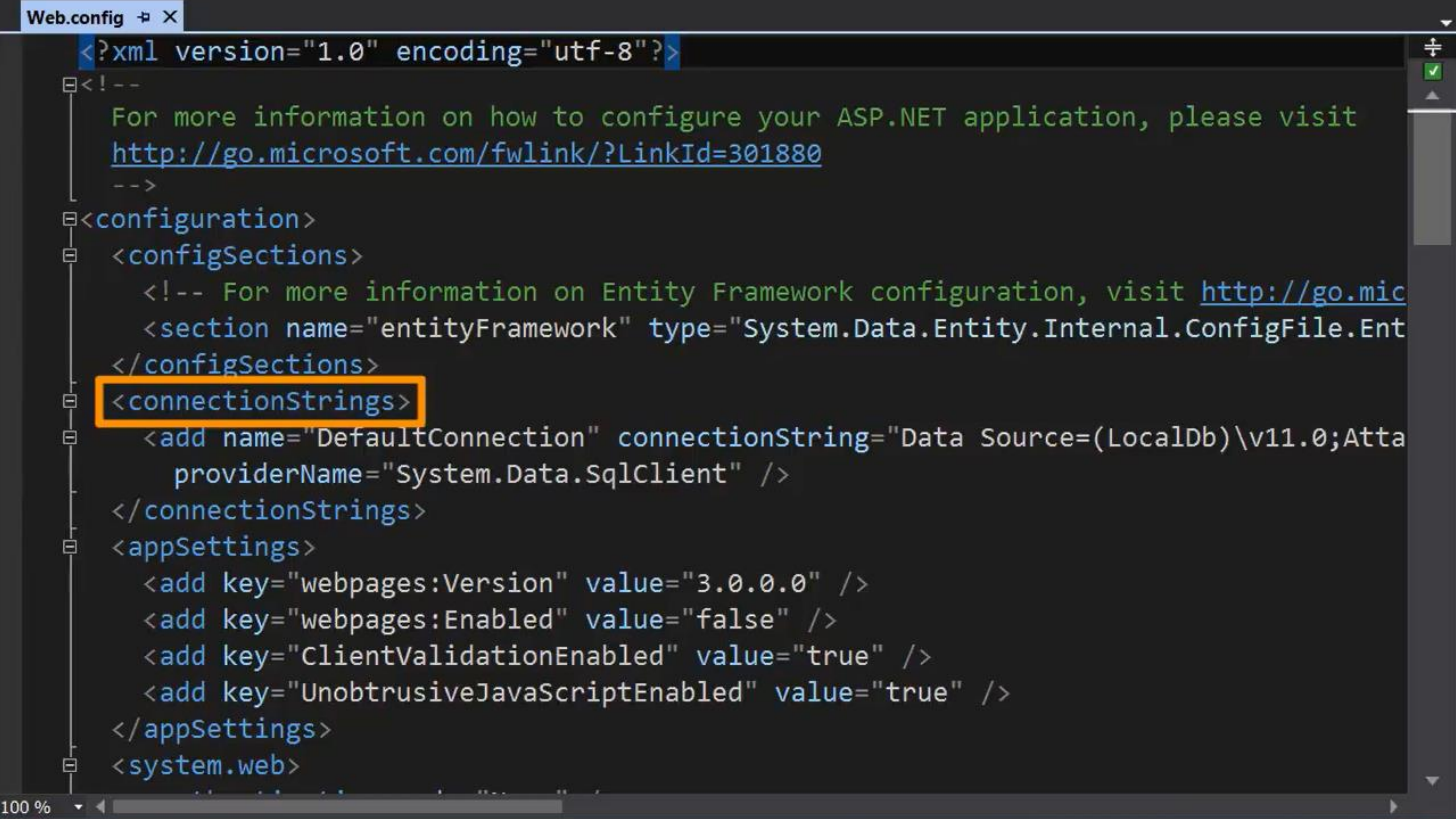
App

`install-package <name>`

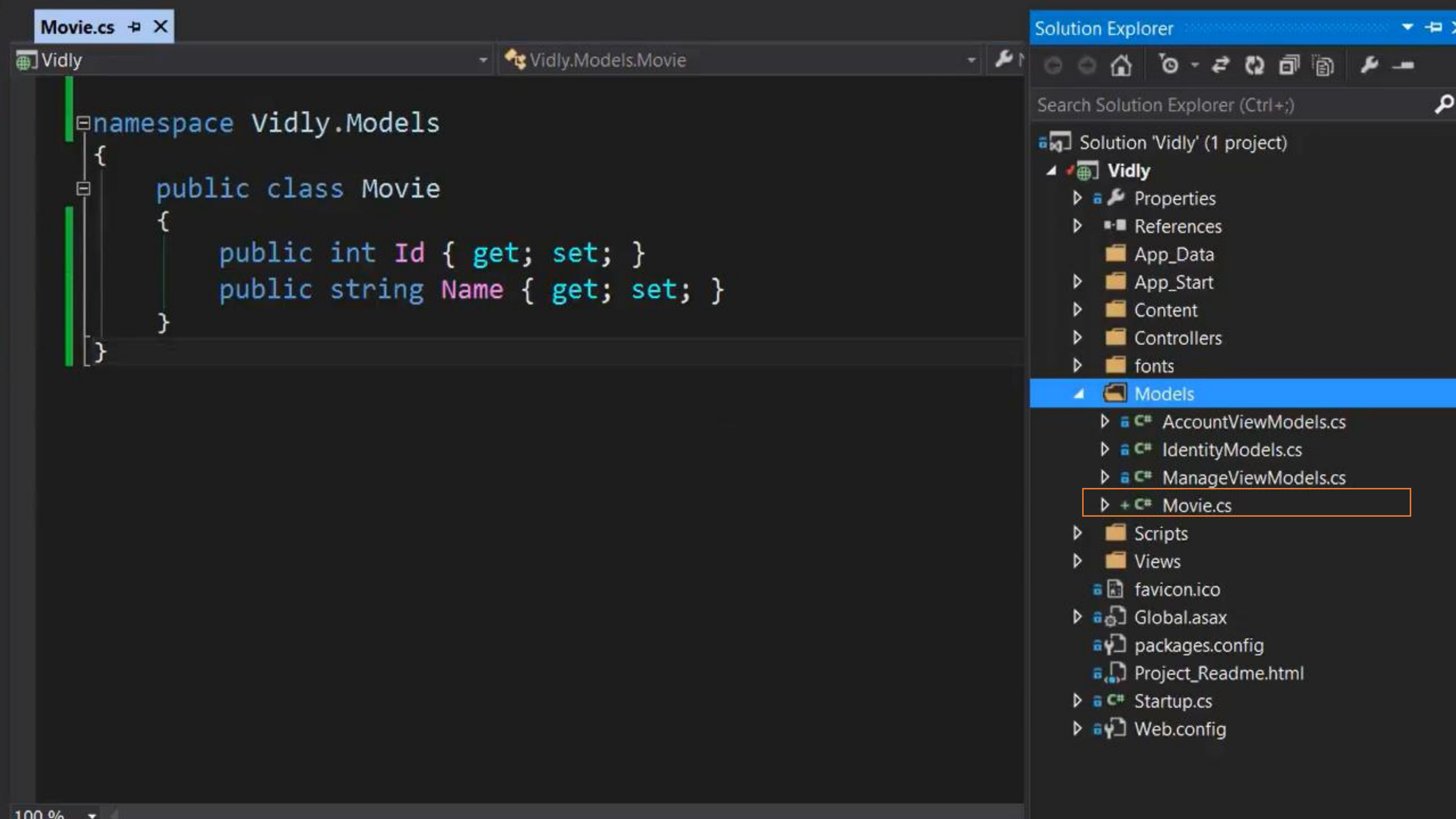
Package Manager

Package1

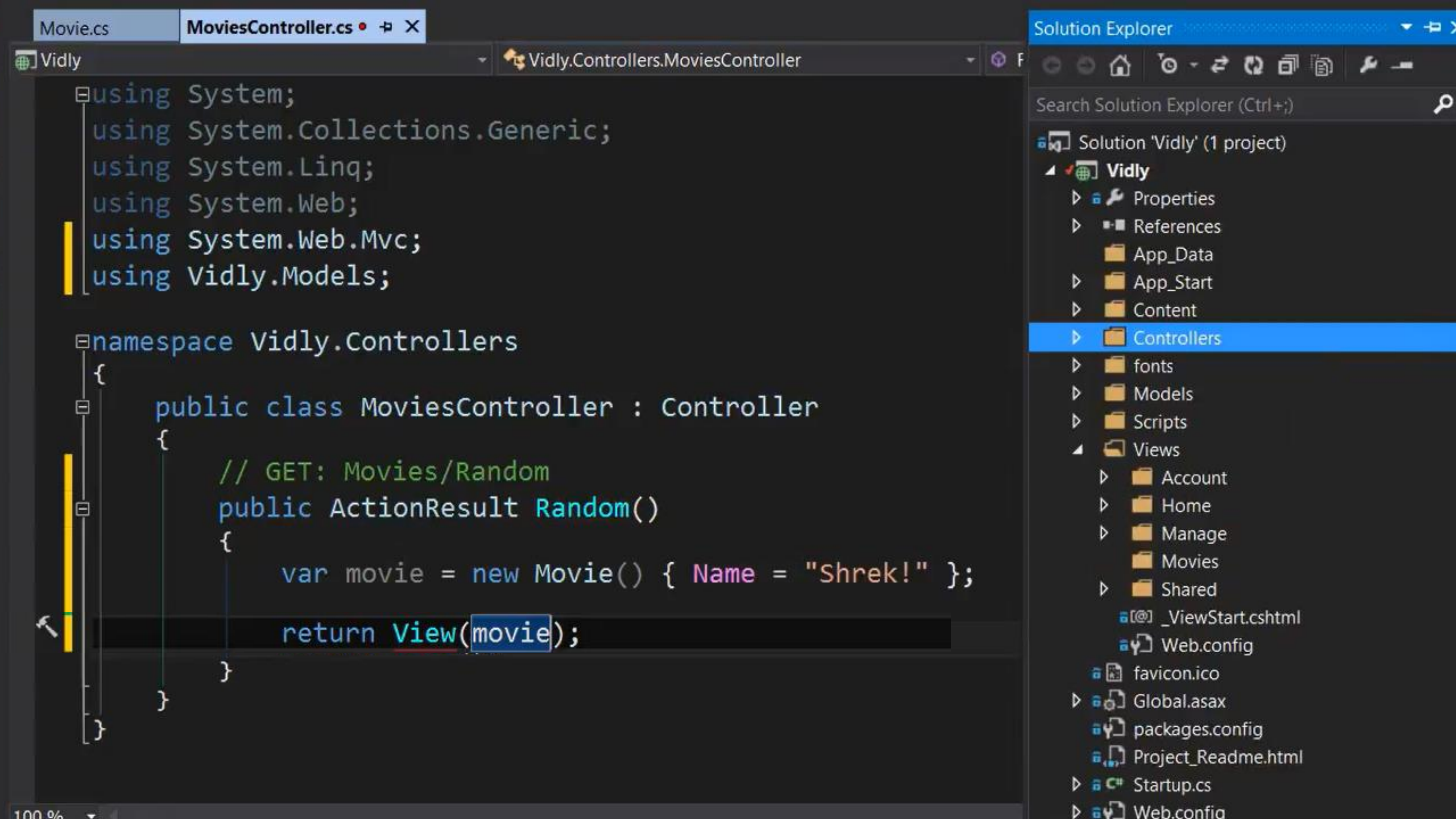
Package2



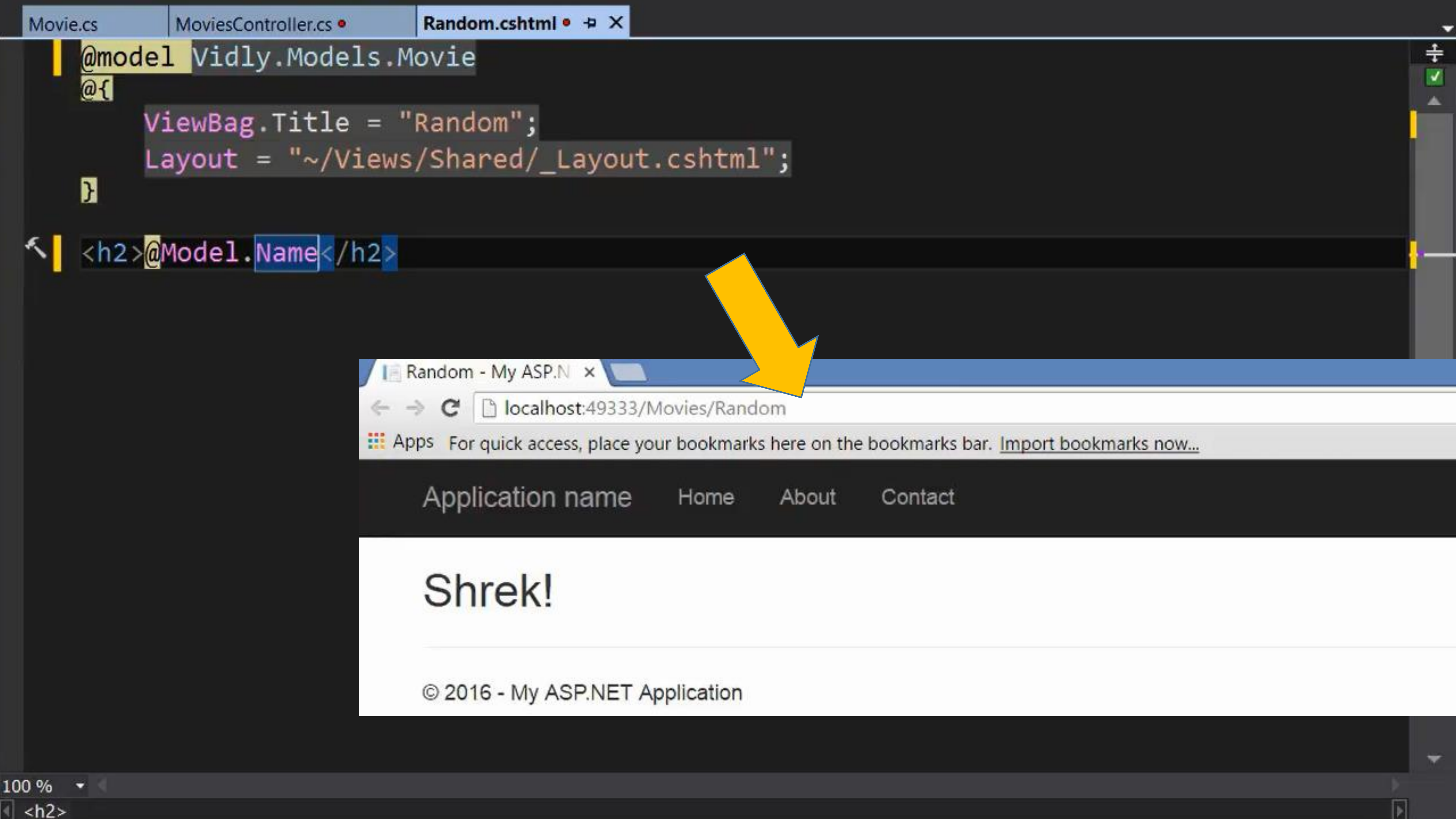
```
<?xml version="1.0" encoding="utf-8"?>
<!--
  For more information on how to configure your ASP.NET application, please visit
  http://go.microsoft.com/fwlink/?LinkId=301880
  -->
<configuration>
  <configSections>
    <!-- For more information on Entity Framework configuration, visit http://go.mic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



```
namespace Vidly.Models
{
    public class Movie
    {
        public int Id { get; set; }
        public string Name { get; set; }
    }
}
```



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using Vidly.Models;

namespace Vidly.Controllers
{
    public class MoviesController : Controller
    {
        // GET: Movies/Random
        public ActionResult Random()
        {
            var movie = new Movie() { Name = "Shrek!" };

            return View(movie);
        }
    }
}
```

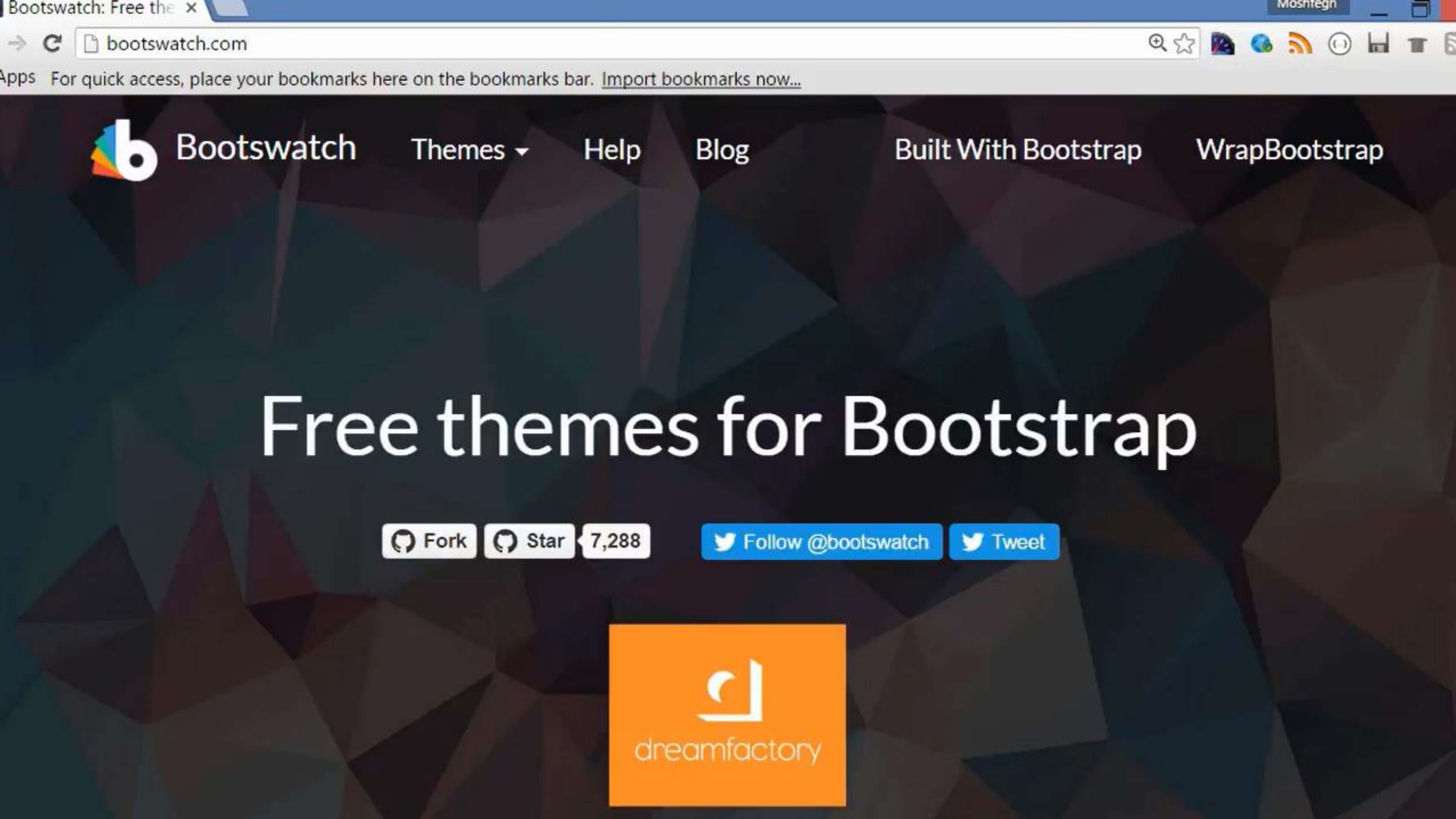

Movie.cs
MoviesController.cs
Random.cshtml
100 %
<h2>

```
@model Vidly.Models.Movie
@{
    ViewBag.Title = "Random";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<h2>@Model.Name</h2>
```

Random - My ASP.N
localhost:49333/Movies/Random
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Application name
Home
About
Contact

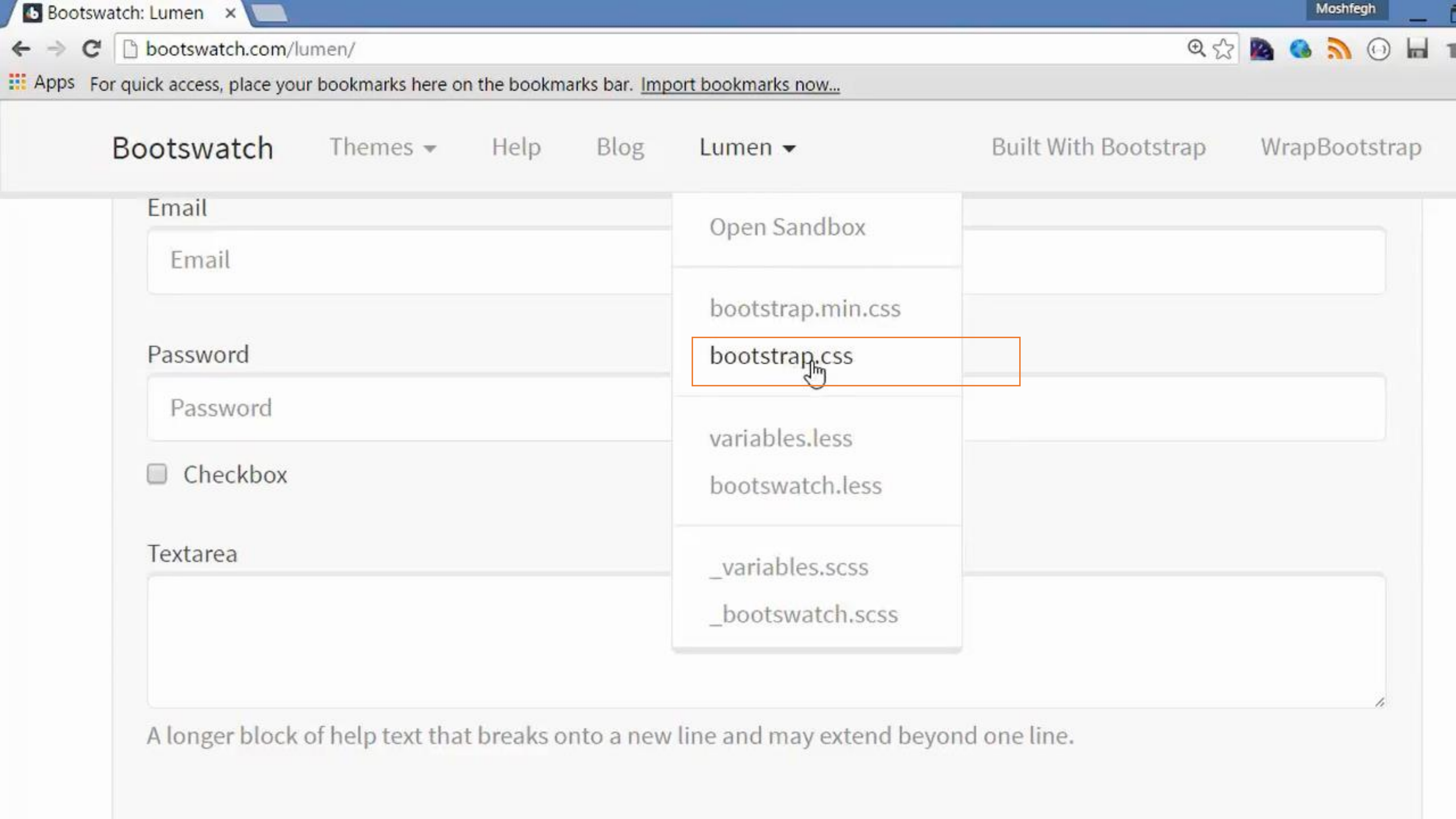
Shrek!

© 2016 - My ASP.NET Application

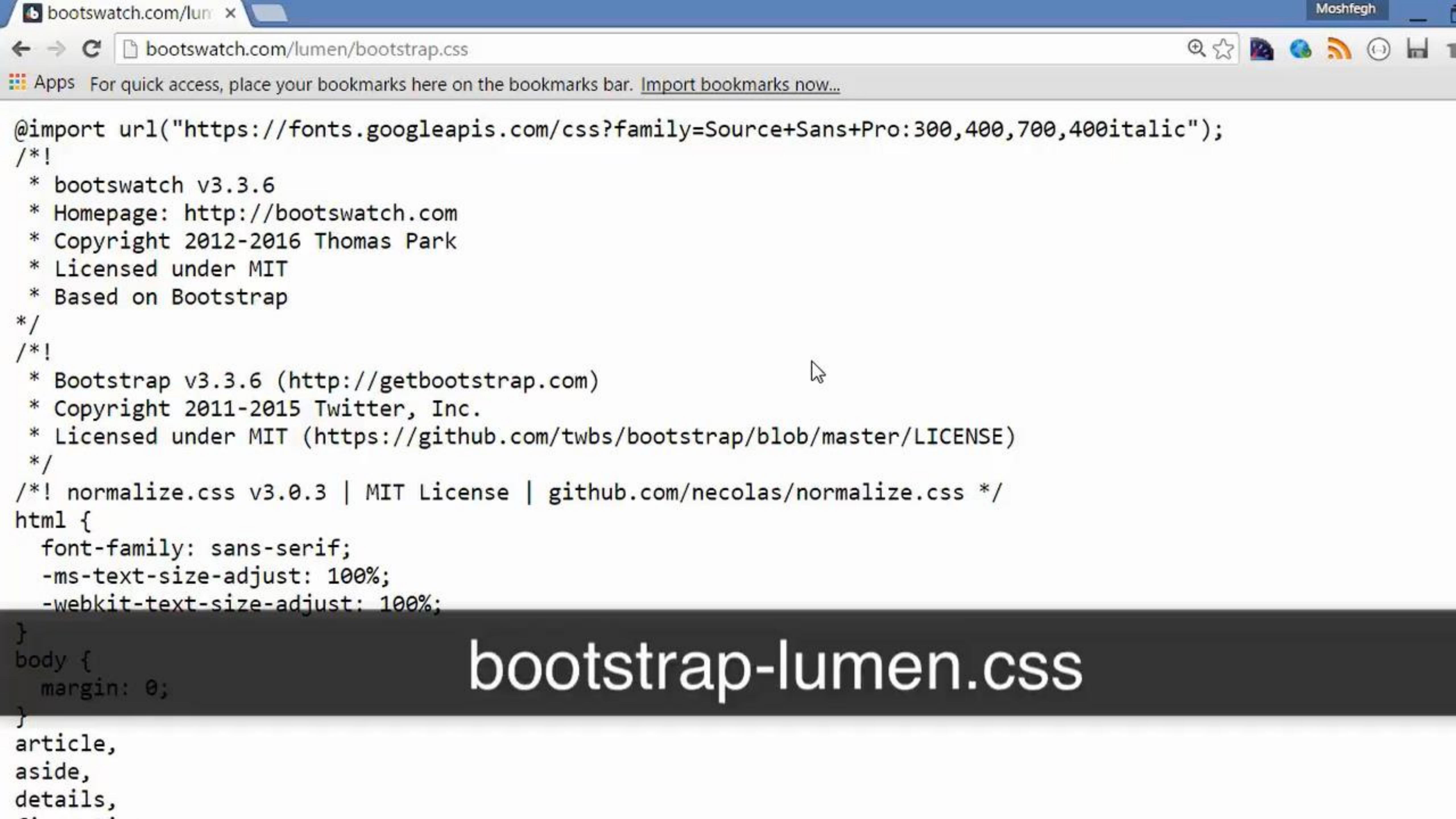


For quick access, place your bookmarks here on the bookmarks bar.

Free themes for Bootstrap

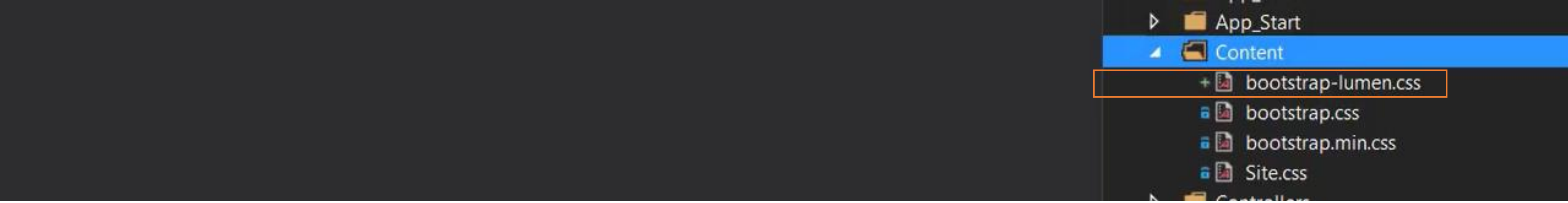


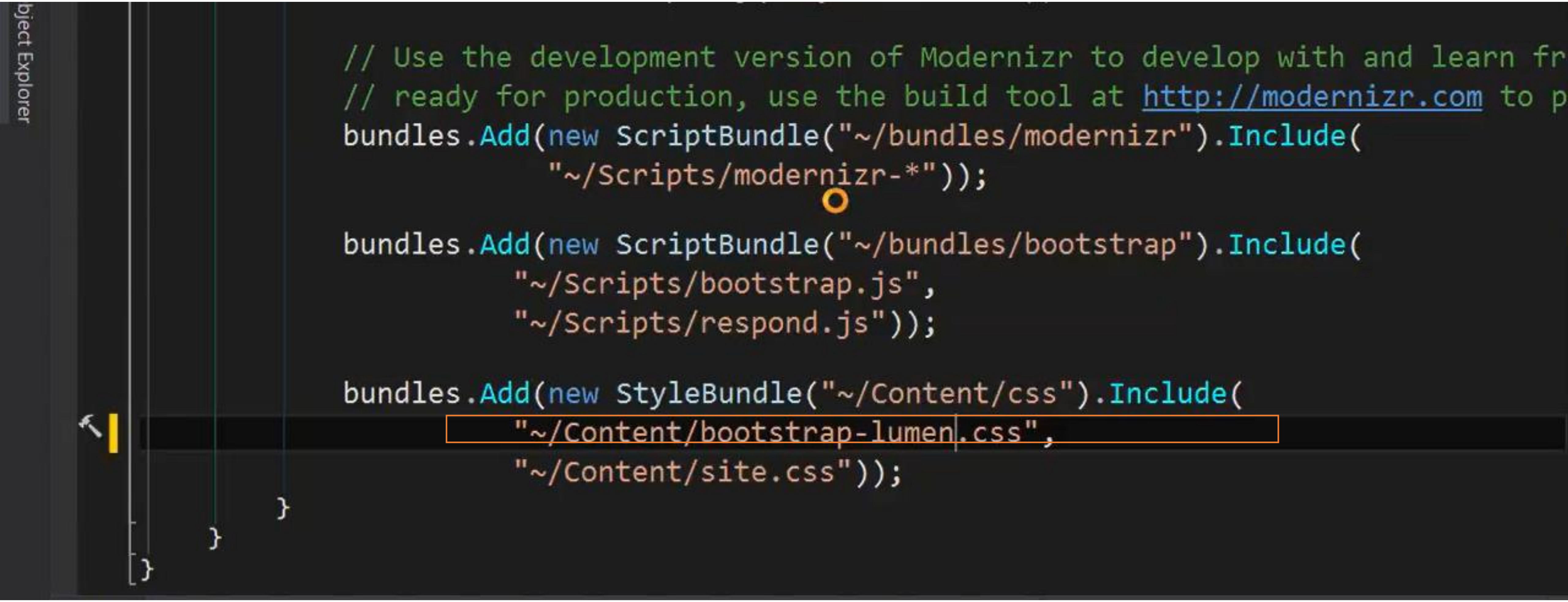
A longer block of help text that breaks onto a new line and may extend beyond one line.


bootswatch.com/lun ×
Moshfegh
bootswatch.com/lumen/bootstrap.css
Apps
For quick access, place your bookmarks here on the bookmarks bar.
Import bookmarks now...

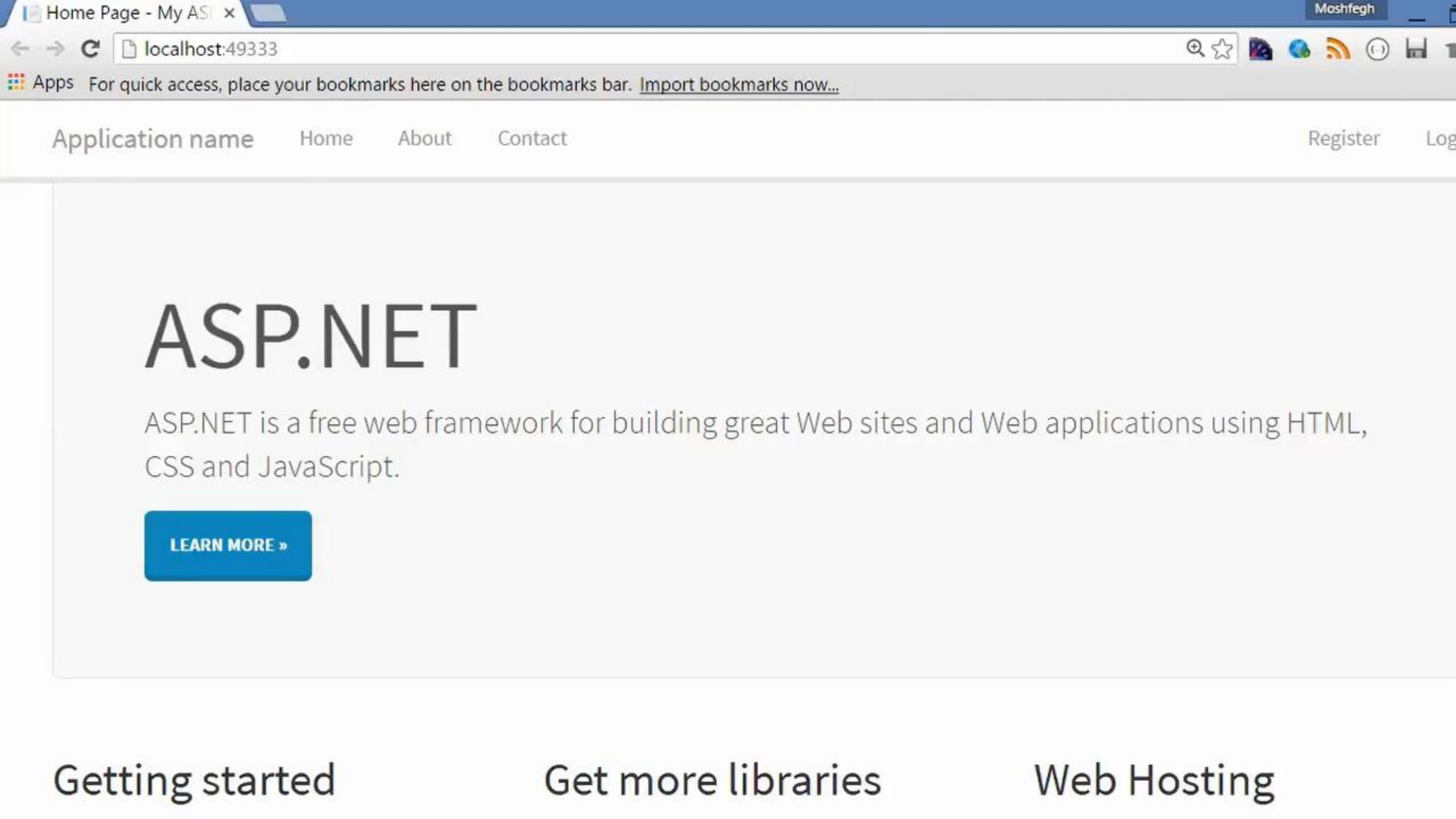
```
@import url("https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,700,400italic");
/*!
 * bootswatch v3.3.6
 * Homepage: http://bootswatch.com
 * Copyright 2012-2016 Thomas Park
 * Licensed under MIT
 * Based on Bootstrap
*/
/*!
 * Bootstrap v3.3.6 (http://getbootstrap.com)
 * Copyright 2011-2015 Twitter, Inc.
 * Licensed under MIT (https://github.com/twbs/bootstrap/blob/master/LICENSE)
 */
/*! normalize.css v3.0.3 | MIT License | github.com/necolas/normalize.css */
html {
  font-family: sans-serif;
  -ms-text-size-adjust: 100%;
  -webkit-text-size-adjust: 100%;
}
body {
  margin: 0;
}
article,
aside,
details,
```

bootstrap-lumen.css





```
            // Use the development version of Modernizr to develop with and learn fr
            // ready for production, use the build tool at http://modernizr.com to p
            bundles.Add(new ScriptBundle("~/bundles/modernizr").Include(
                        "~/Scripts/modernizr-*"));

            bundles.Add(new ScriptBundle("~/bundles/bootstrap").Include(
                      "~/Scripts/bootstrap.js",
                      "~/Scripts/respond.js"));

            bundles.Add(new StyleBundle("~/Content/css").Include(
                      "~/Content/bootstrap-lumen.css",
                      "~/Content/site.css"));
        }
    }
}
```



ASP.NET

ASP.NET is a free web framework for building great Web sites and Web applications using HTML, CSS and JavaScript.

Getting started

Get more libraries

Web Hosting

Course Layout

Course Layout

- ASP.NET MVC Fundamentals
- Entity Framework (Code-first)
- Forms
- Validation
- Build RESTful Services
- Client-side Development
- Authentication and Authorization
- Performance Optimization
- Building a Feature Systematically

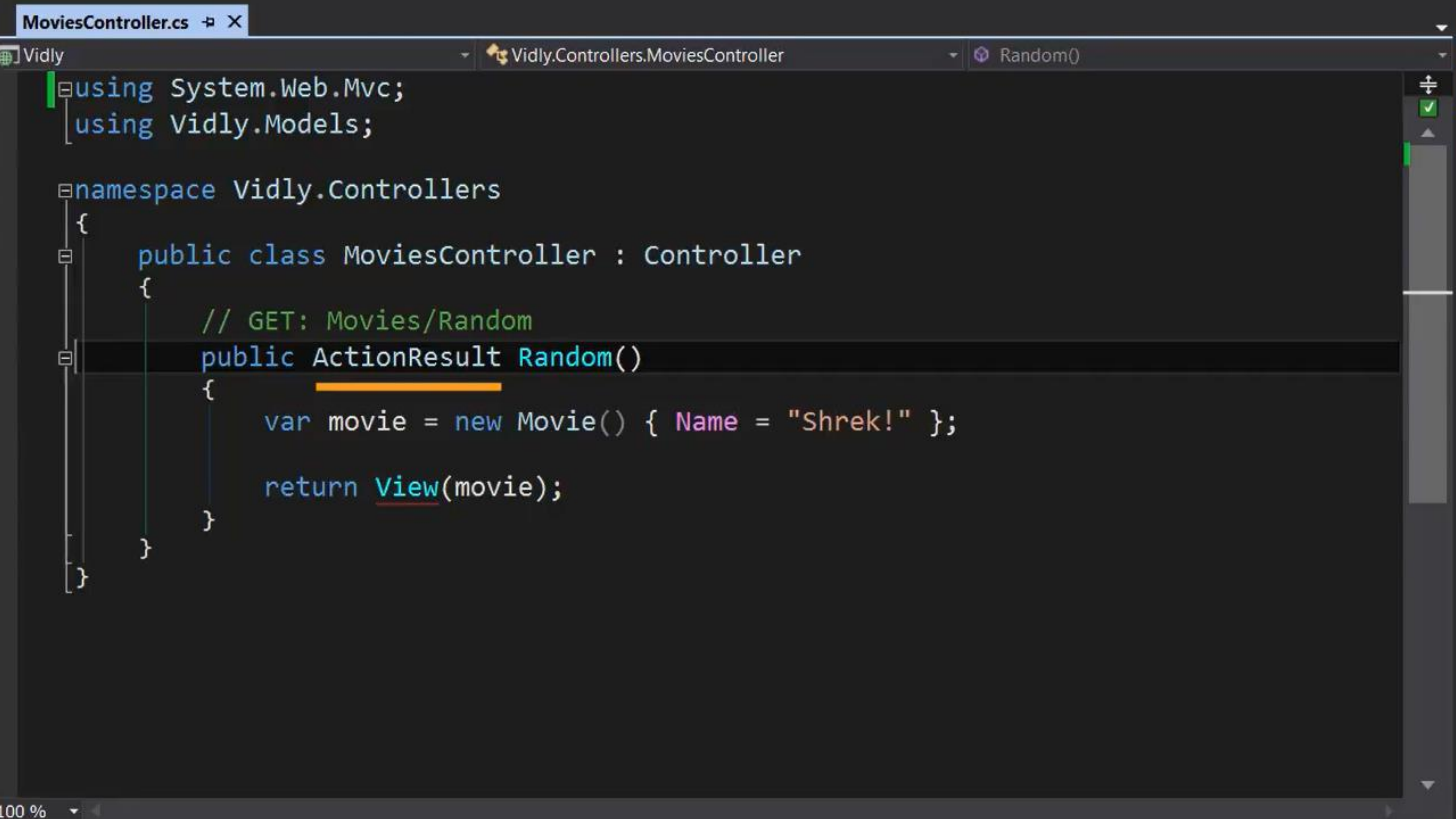
Next

ASP.NET MVC Fundamentals

Action Result

Zouhair Rimale, Ph.D.

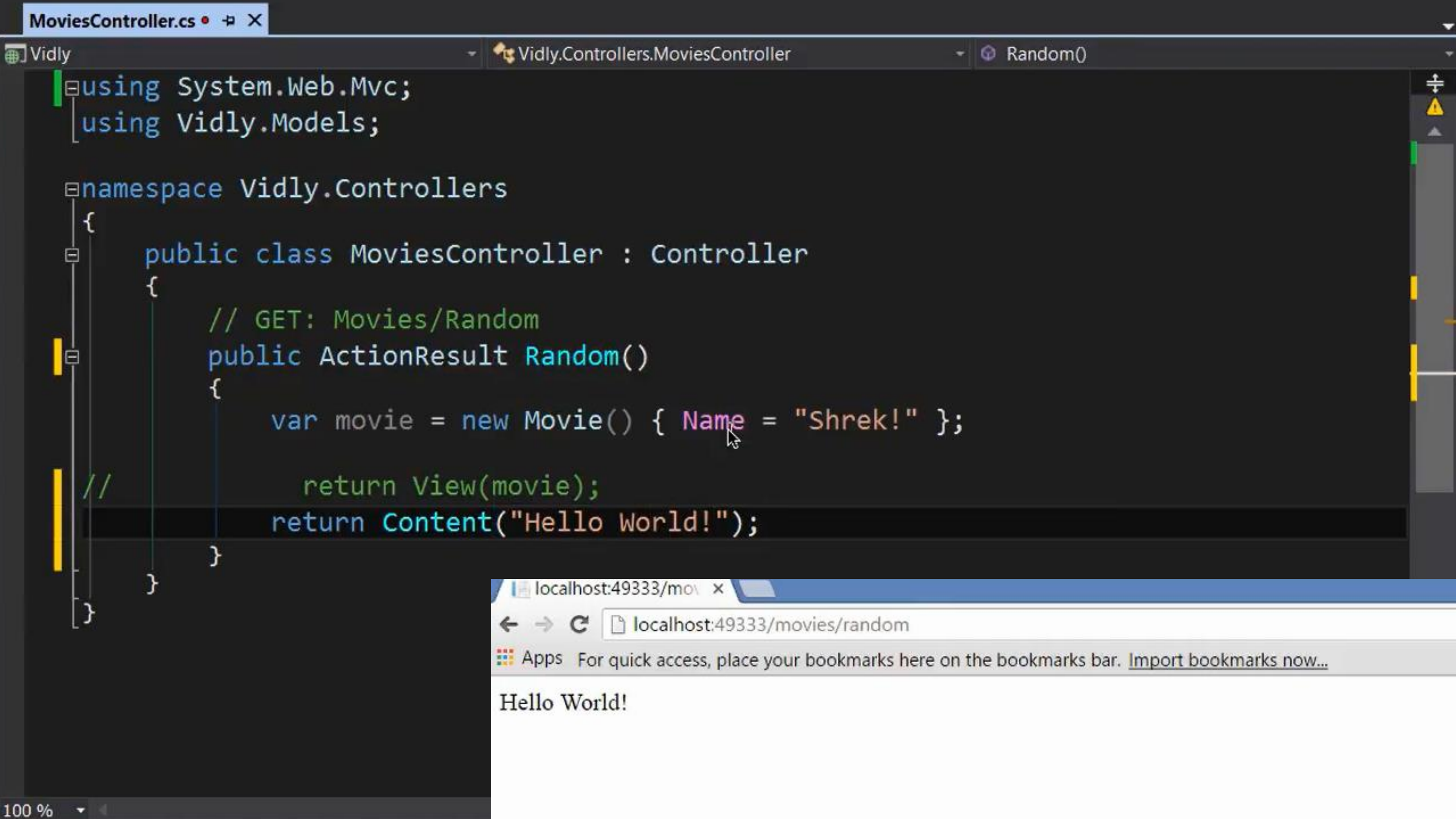
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



```
using System.Web.Mvc;
using Vidly.Models;

namespace Vidly.Controllers
{
    public class MoviesController : Controller
    {
        // GET: Movies/Random
        public ActionResult Random()
        {
            var movie = new Movie() { Name = "Shrek!" };

            return View(movie);
        }
    }
}
```

Action Results

Type	Helper Method
ViewResult	View()
PartialViewResult	PartialView()
ContentResult	Content()
RedirectResult	Redirect()
RedirectToRouteResult	RedirectToAction()
JsonResult	Json()
FileResult	File()
HttpNotFoundResult	HttpNotFound()
EmptyResult	

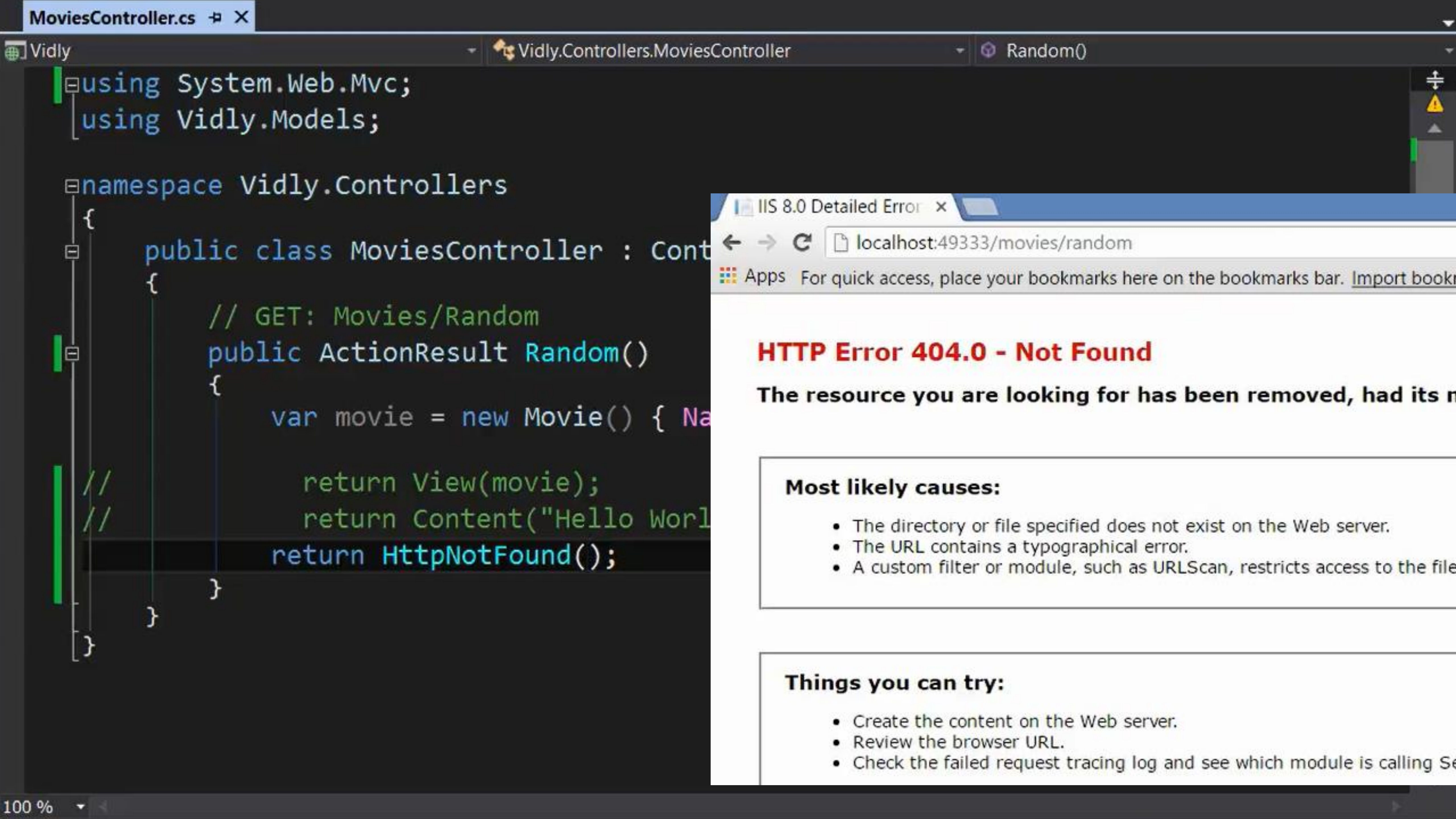


```
using System.Web.Mvc;
using Vidly.Models;

namespace Vidly.Controllers
{
    public class MoviesController : Controller
    {
        // GET: Movies/Random
        public ActionResult Random()
        {
            var movie = new Movie() { Name = "Shrek!" };

//              return View(movie);
            return Content("Hello World!");
        }
    }
}
```

localhost:49333/movies/random

For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

```
Hello World!
```



MoviesController.cs
Vidly
Vidly.Controllers.MoviesController
Random()
100 %

```
using System.Web.Mvc;
using Vidly.Models;

namespace Vidly.Controllers
{
    public class MoviesController : Cont
    {
        // GET: Movies/Random
        public ActionResult Random()
        {
            var movie = new Movie() { Na

//              return View(movie);
//              return Content("Hello Worl
            return HttpNotFound();
        }
    }
}
```

IIS 8.0 Detailed Error
localhost:49333/movies/random
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookr

HTTP Error 404.0 - Not Found

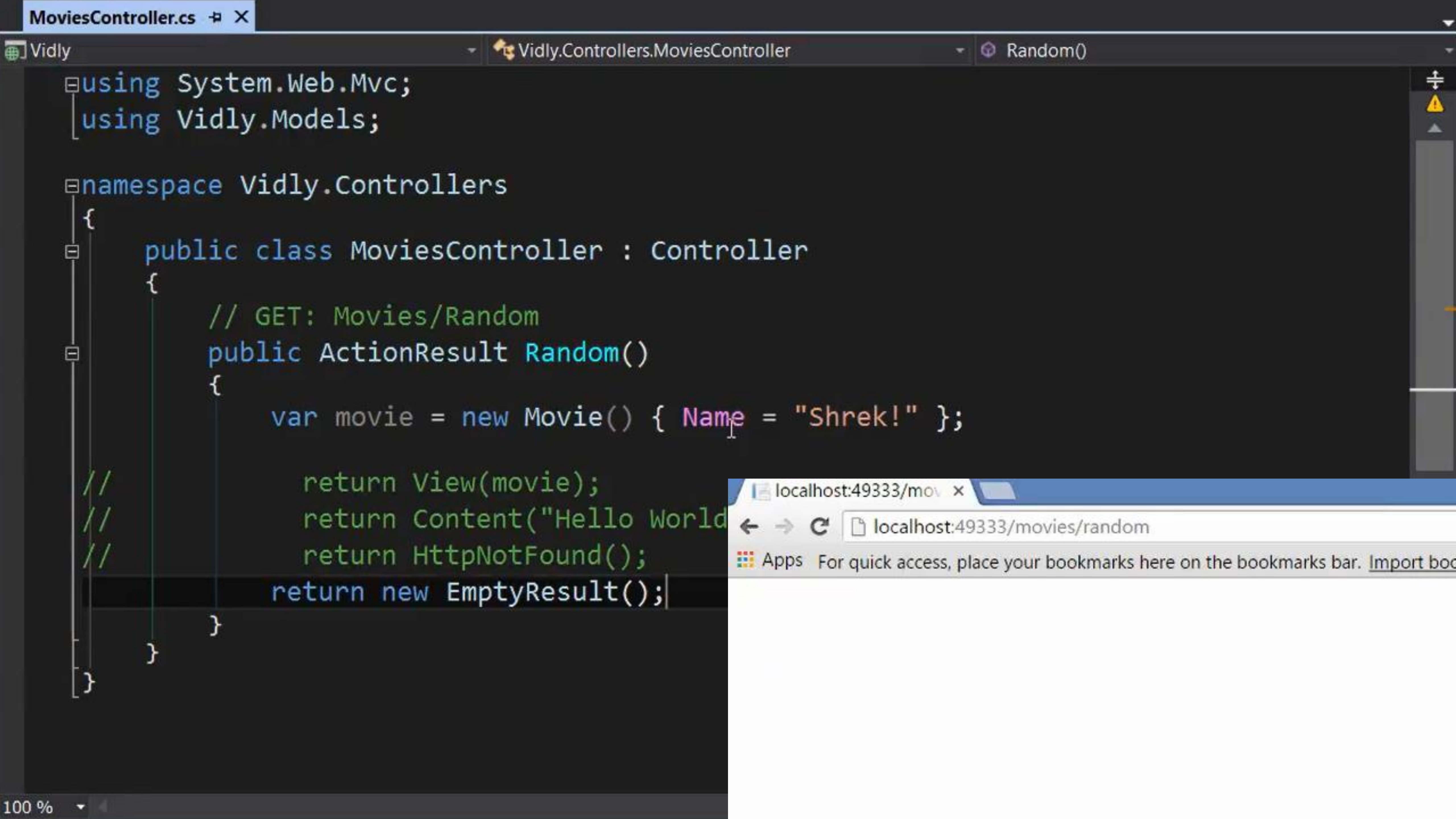
The resource you are looking for has been removed, had its n

Most likely causes:

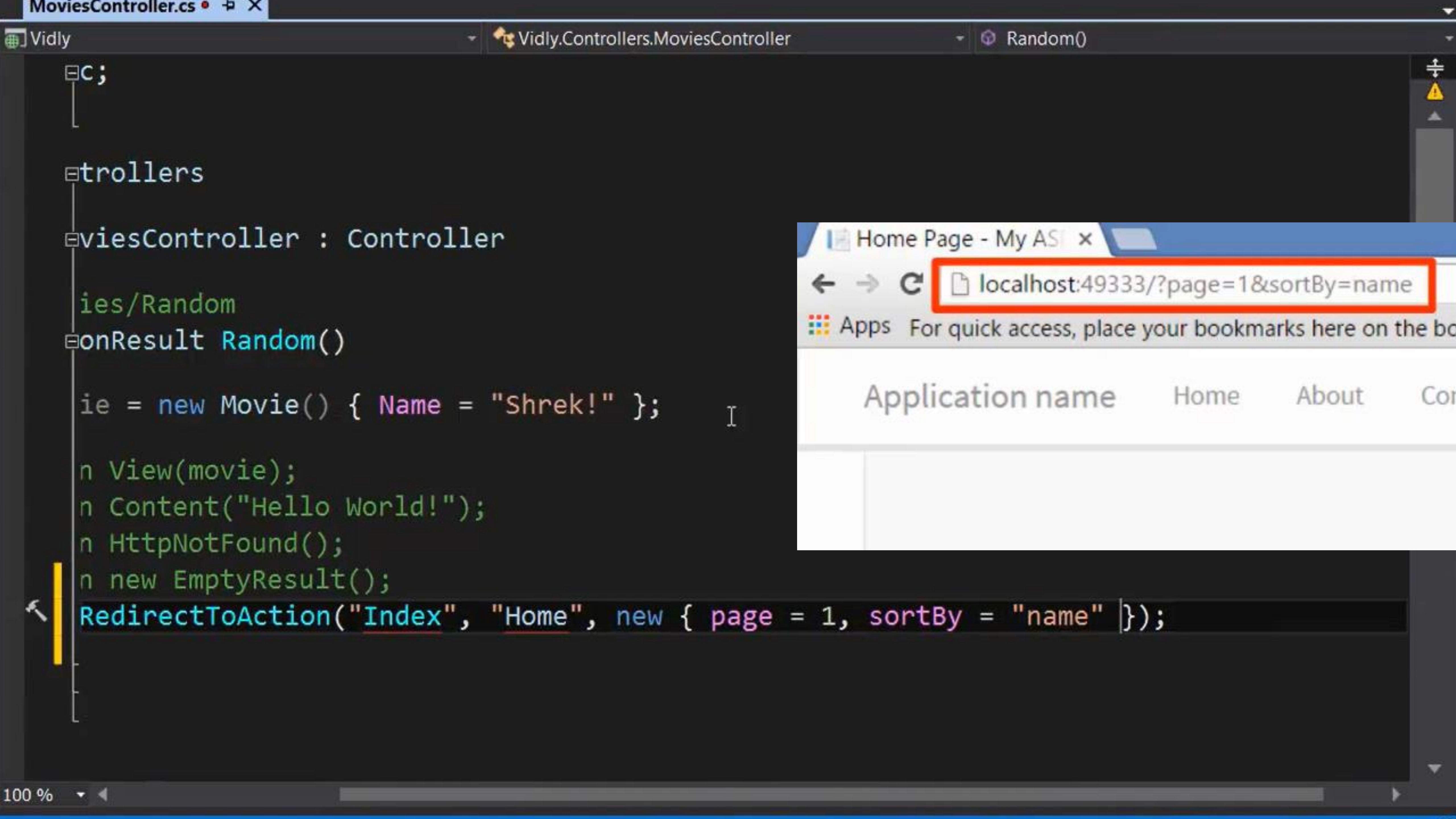
- The directory or file specified does not exist on the Web server.
- The URL contains a typographical error.
- A custom filter or module, such as URLScan, restricts access to the file

Things you can try:

- Create the content on the Web server.
- Review the browser URL.
- Check the failed request tracing log and see which module is calling Se


MoviesController.cs
Vidly
Vidly.Controllers.MoviesController
Random()
100 %
localhost:49333/mo
localhost:49333/movies/random
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import boo

```
using System.Web.Mvc;
using Vidly.Models;

namespace Vidly.Controllers
{
    public class MoviesController : Controller
    {
        // GET: Movies/Random
        public ActionResult Random()
        {
            var movie = new Movie() { Name = "Shrek!" };

//              return View(movie);
//              return Content("Hello World
//              return HttpNotFound();
            return new EmptyResult();
        }
    }
}
```



MoviesController.cs
Vidly
Vidly.Controllers.MoviesController
Random()
100 %

```
C;

trollers

viesController : Controller

ies/Random
onResult Random()

ie = new Movie() { Name = "Shrek!" };

n View(movie);
n Content("Hello World!");
n HttpNotFound();
n new EmptyResult();
RedirectToAction("Index", "Home", new { page = 1, sortBy = "name" });
```

Home Page - My AS
Apps
For quick access, place your bookmarks here on the bo
Application name
Home
About
Co

localhost:49333/?page=1&sortBy=name

Action Results

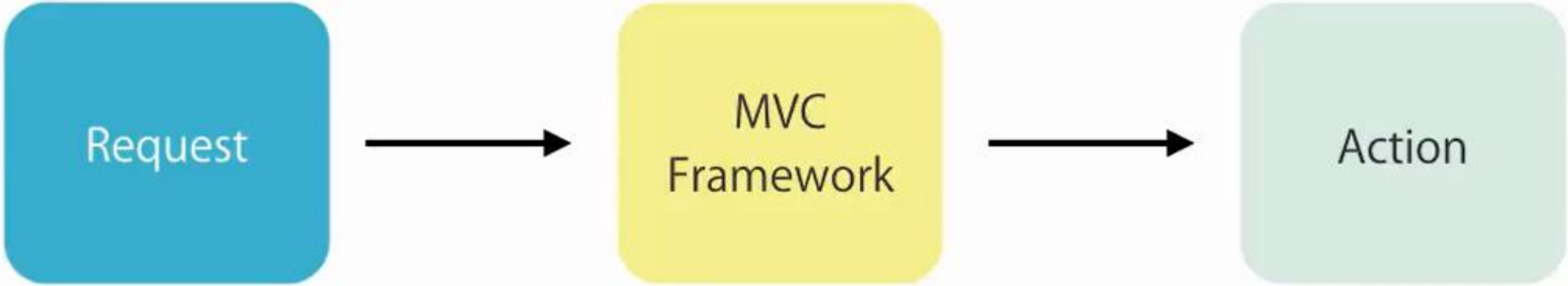
Type	Helper Method
ViewResult	View()
PartialViewResult	PartialView()
ContentResult	Content()
RedirectResult	Redirect()
RedirectToRouteResult	RedirectToAction()
JsonResult	Json()
FileResult	File()
HttpNotFoundResult	HttpNotFound()
EmptyResult	

Action Parameters

Zouhair Rimale, Ph.D.

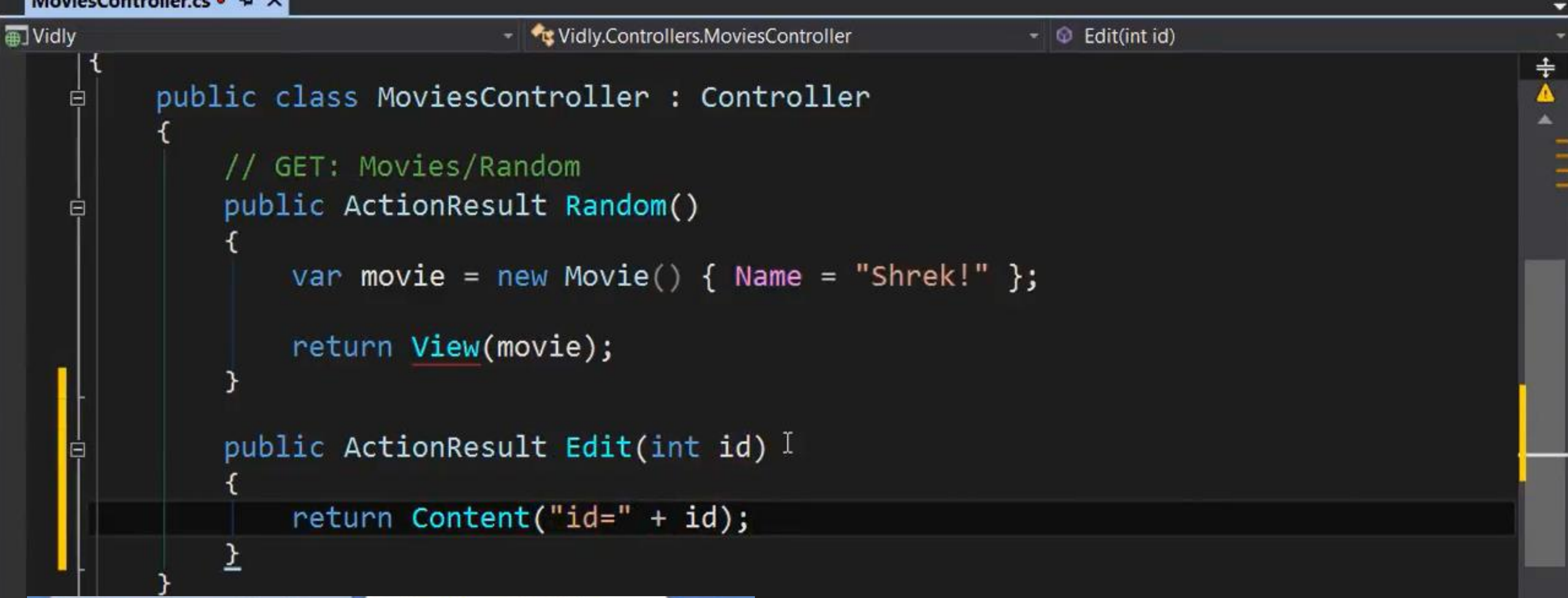
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular

Parameter Binding

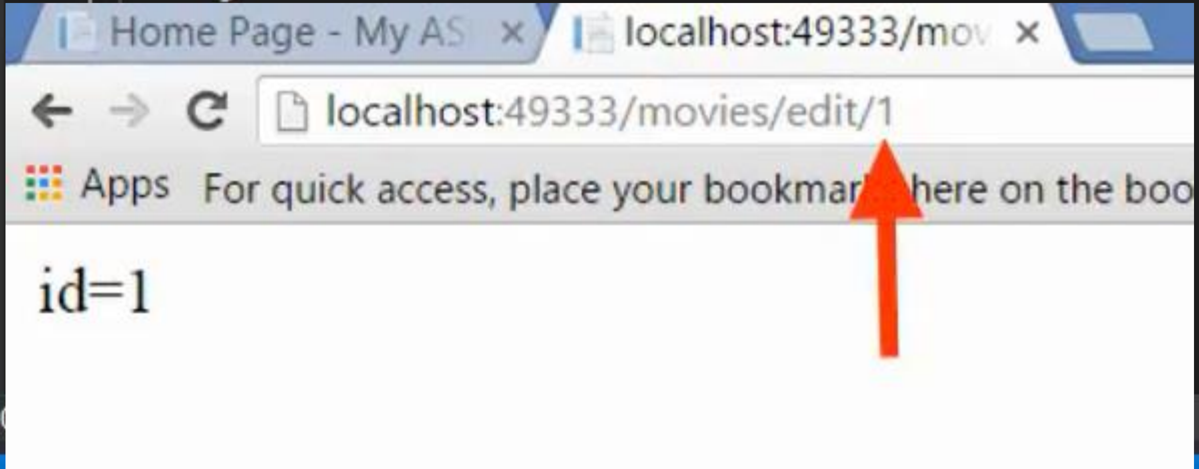


Parameter Sources

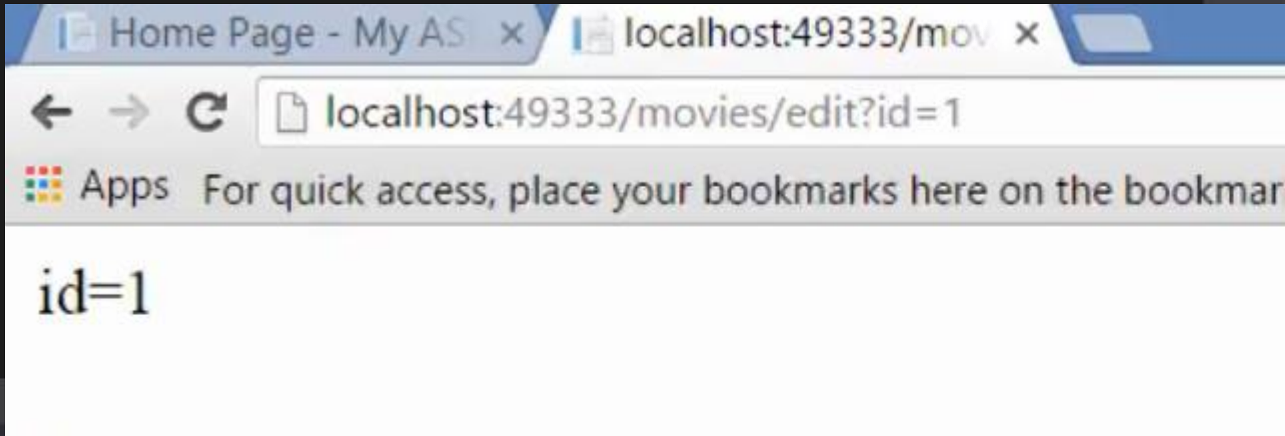
- In the URL: `/movies/edit/1`
- In the query string: `/movies/edit?id=1`
- In the form data: `id=1`



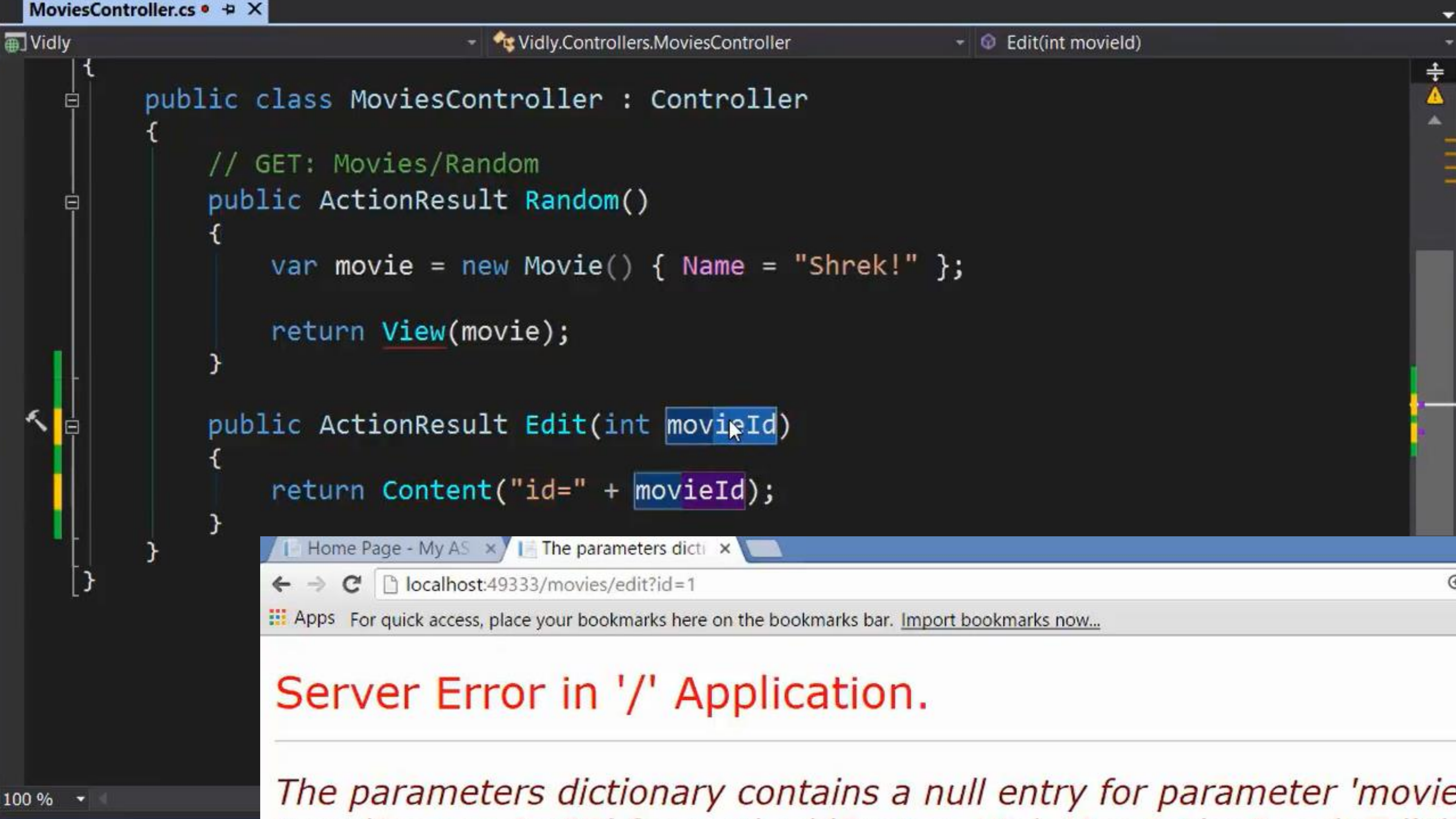
```
{
    public class MoviesController : Controller
    {
        // GET: Movies/Random
        public ActionResult Random()
        {
            var movie = new Movie() { Name = "Shrek!" };

            return View(movie);
        }

        public ActionResult Edit(int id)
        {
            return Content("id=" + id);
        }
    }
```


Home Page - My AS
localhost:49333/mov
localhost:49333/movies/edit/1
Apps
For quick access, place your bookmar here on the boo

id=1


Home Page - My AS
localhost:49333/mov
localhost:49333/movies/edit?id=1
Apps
For quick access, place your bookmarks here on the bookmar

id=1

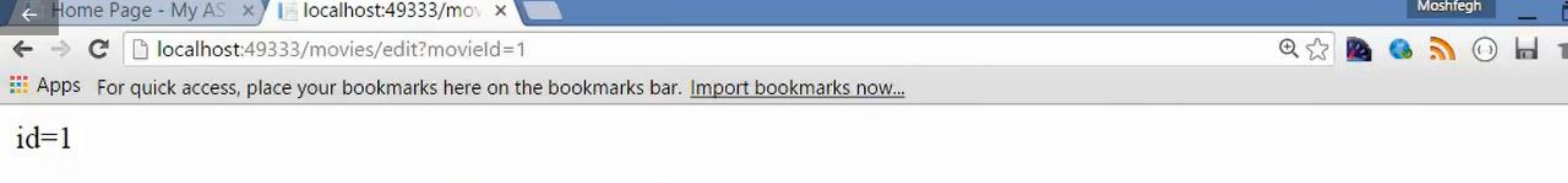


```
{
    public class MoviesController : Controller
    {
        // GET: Movies/Random
        public ActionResult Random()
        {
            var movie = new Movie() { Name = "Shrek!" };

            return View(movie);
        }

        public ActionResult Edit(int movieId)
        {
            return Content("id=" + movieId);
        }
    }
}
```

localhost:49333/movies/edit?id=1

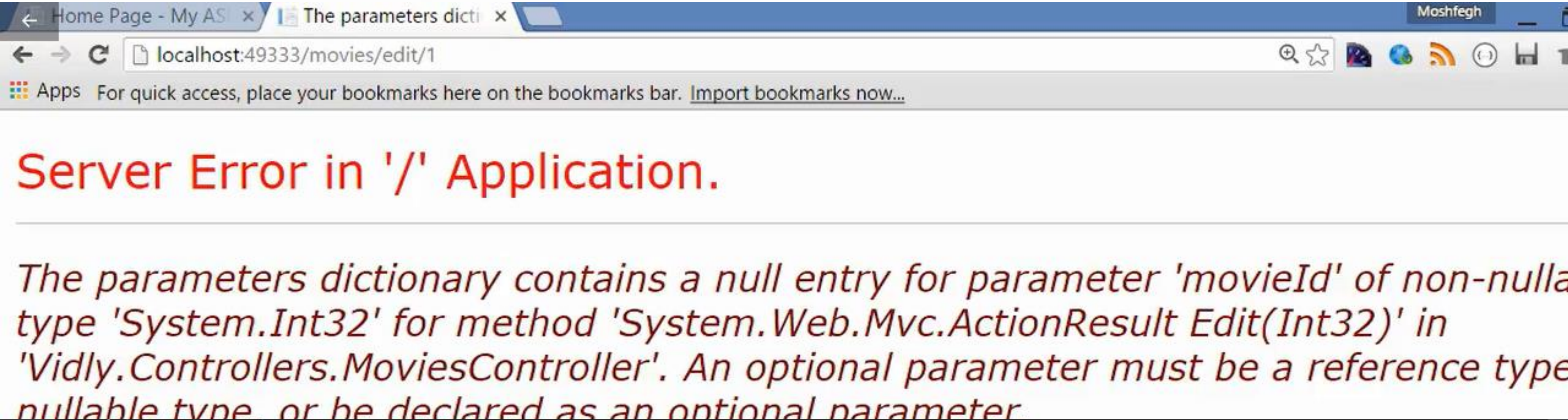
Server Error in '/' Application.

The parameters dictionary contains a null entry for parameter 'movie



Home Page - My AS
localhost:49333/mo
Moshfegh
localhost:49333/movies/edit?movieId=1
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

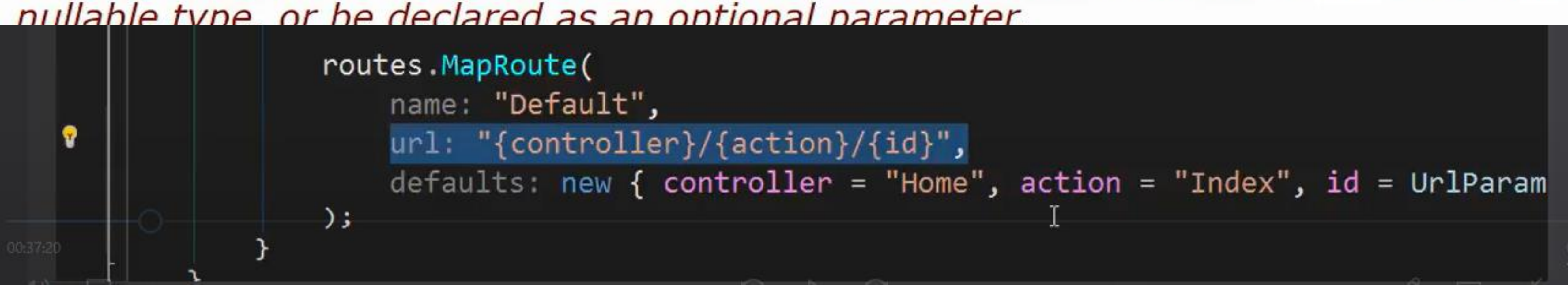
id=1



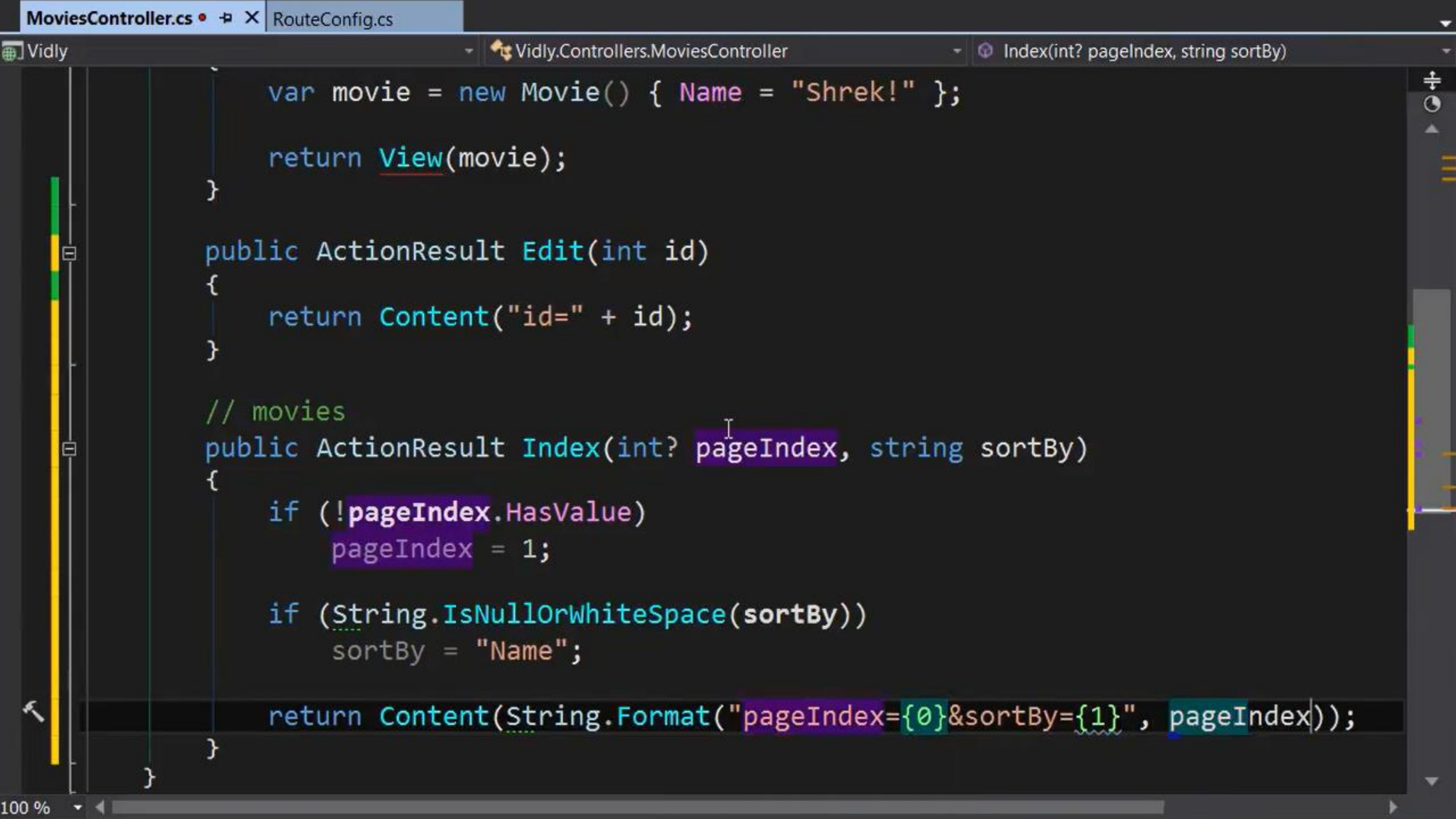
Home Page - My AS
The parameters dicti
Moshfegh
localhost:49333/movies/edit/1
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

Server Error in '/' Application.

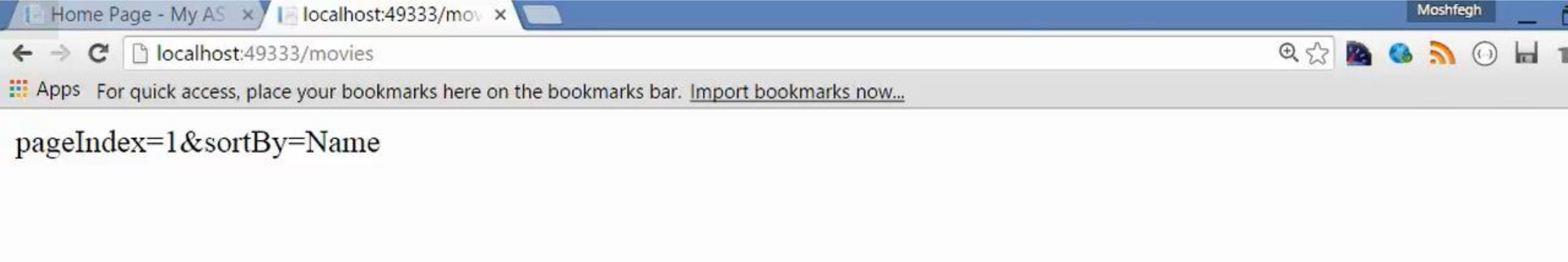
The parameters dictionary contains a null entry for parameter 'movieId' of non-nulla type 'System.Int32' for method 'System.Web.Mvc.ActionResult Edit(Int32)' in 'Vidly.Controllers.MoviesController'. An optional parameter must be a reference type nullable type, or be declared as an optional parameter.



```
            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Home", action = "Index", id = UrlParam
            );
        }
```

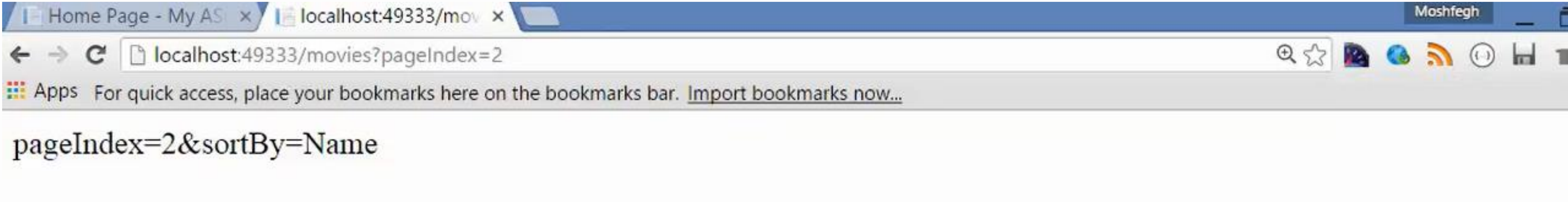

MoviesController.cs
RouteConfig.cs
Vidly
Vidly.Controllers.MoviesController
Index(int? pageIndex, string sortBy)
100 %

```
            var movie = new Movie() { Name = "Shrek!" };

            return View(movie);
        }

        public ActionResult Edit(int id)
        {
            return Content("id=" + id);
        }

        // movies
        public ActionResult Index(int? pageIndex, string sortBy)
        {
            if (!pageIndex.HasValue)
                pageIndex = 1;

            if (String.IsNullOrWhiteSpace(sortBy))
                sortBy = "Name";

            return Content(String.Format("pageIndex={0}&sortBy={1}", pageIndex));
        }
    }
```


Home Page - My AS
localhost:49333/mov
Moshfegh
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

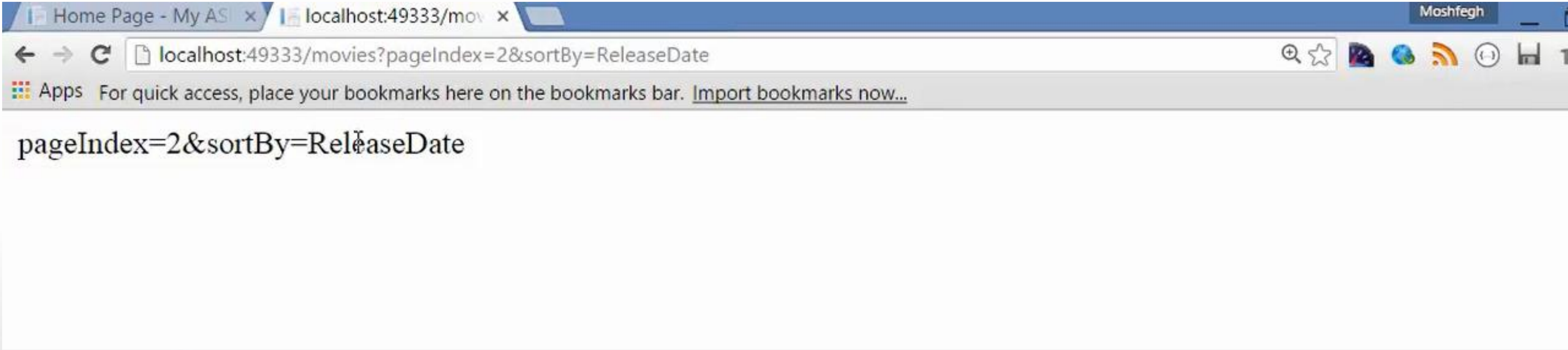
localhost:49333/movies

pageIndex=1&sortBy=Name


Home Page - My AS
localhost:49333/mov
Moshfegh
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

localhost:49333/movies?pageIndex=2

pageIndex=2&sortBy=Name


Home Page - My AS
localhost:49333/mov
Moshfegh
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

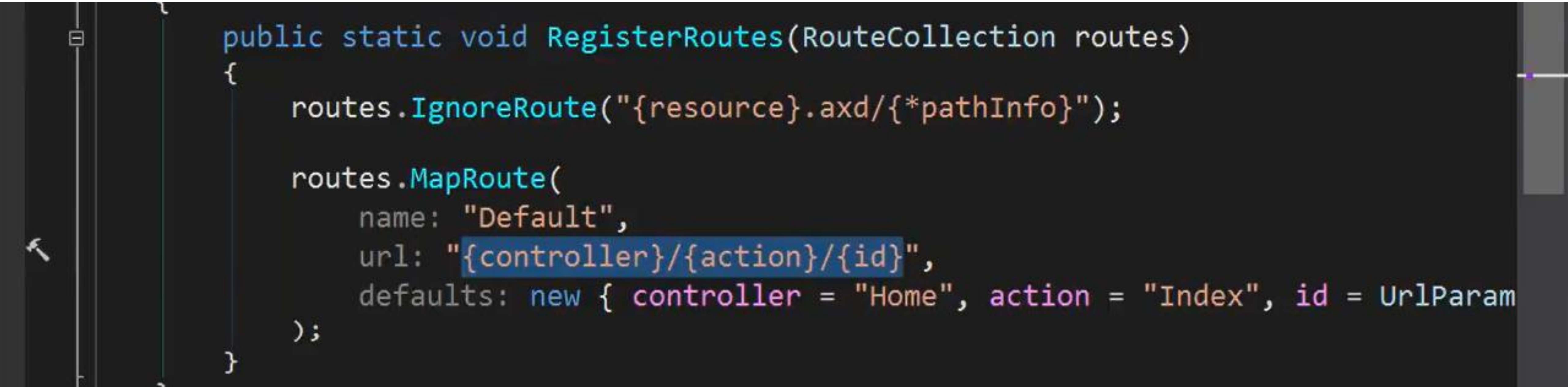
localhost:49333/movies?pageIndex=2&sortBy=ReleaseDate

pageIndex=2&sortBy=ReleaseDate

Convention-based Routes

Zouhair Rimale, Ph.D.

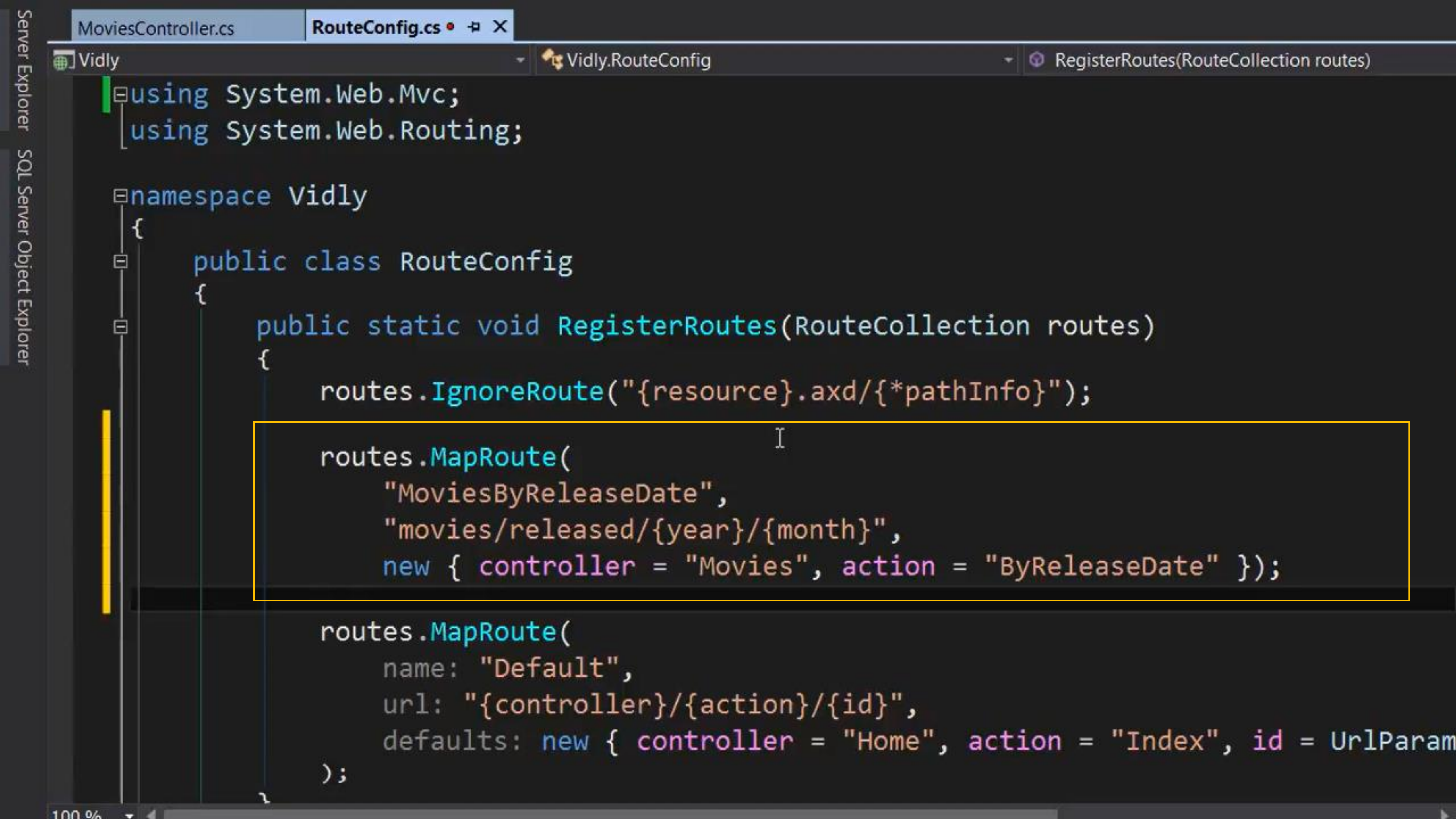
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



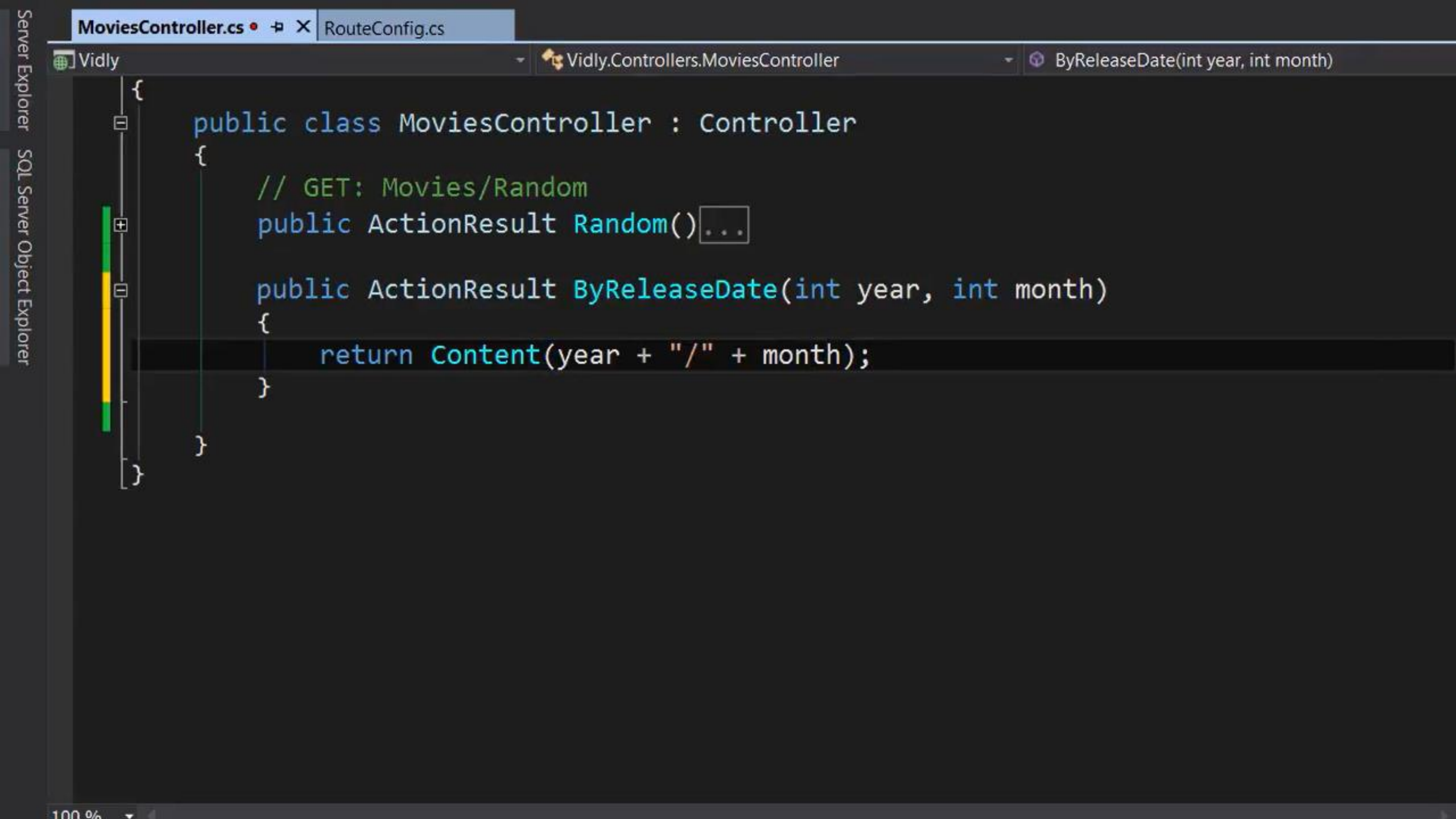
```
public static void RegisterRoutes(RouteCollection routes)
{
    routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

    routes.MapRoute(
        name: "Default",
        url: "{controller}/{action}/{id}",
        defaults: new { controller = "Home", action = "Index", id = UrlParam
    );
}
```

Custom Route

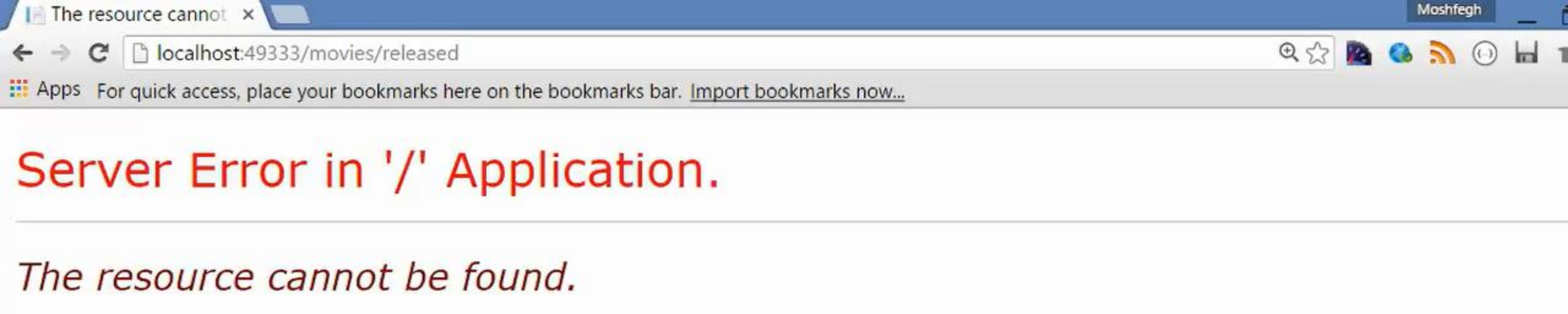
```
/movies/released/2015/04
```



```
using System.Web.Mvc;
using System.Web.Routing;

namespace Vidly
{
    public class RouteConfig
    {
        public static void RegisterRoutes(RouteCollection routes)
        {
            routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

            routes.MapRoute(
                "MoviesByReleaseDate",
                "movies/released/{year}/{month}",
                new { controller = "Movies", action = "ByReleaseDate" });

            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Home", action = "Index", id = UrlParam
            );
        }
```

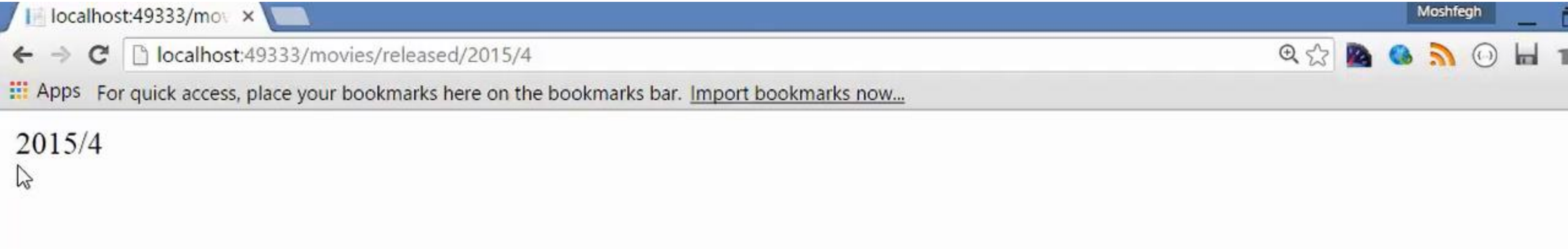

MoviesController.cs
RouteConfig.cs
Vidly
Vidly.Controllers.MoviesController
ByReleaseDate(int year, int month)
Server Explorer
SQL Server Object Explorer
100 %

```
{
    public class MoviesController : Controller
    {
        // GET: Movies/Random
        public ActionResult Random()...

        public ActionResult ByReleaseDate(int year, int month)
        {
            return Content(year + "/" + month);
        }

    }
}
```

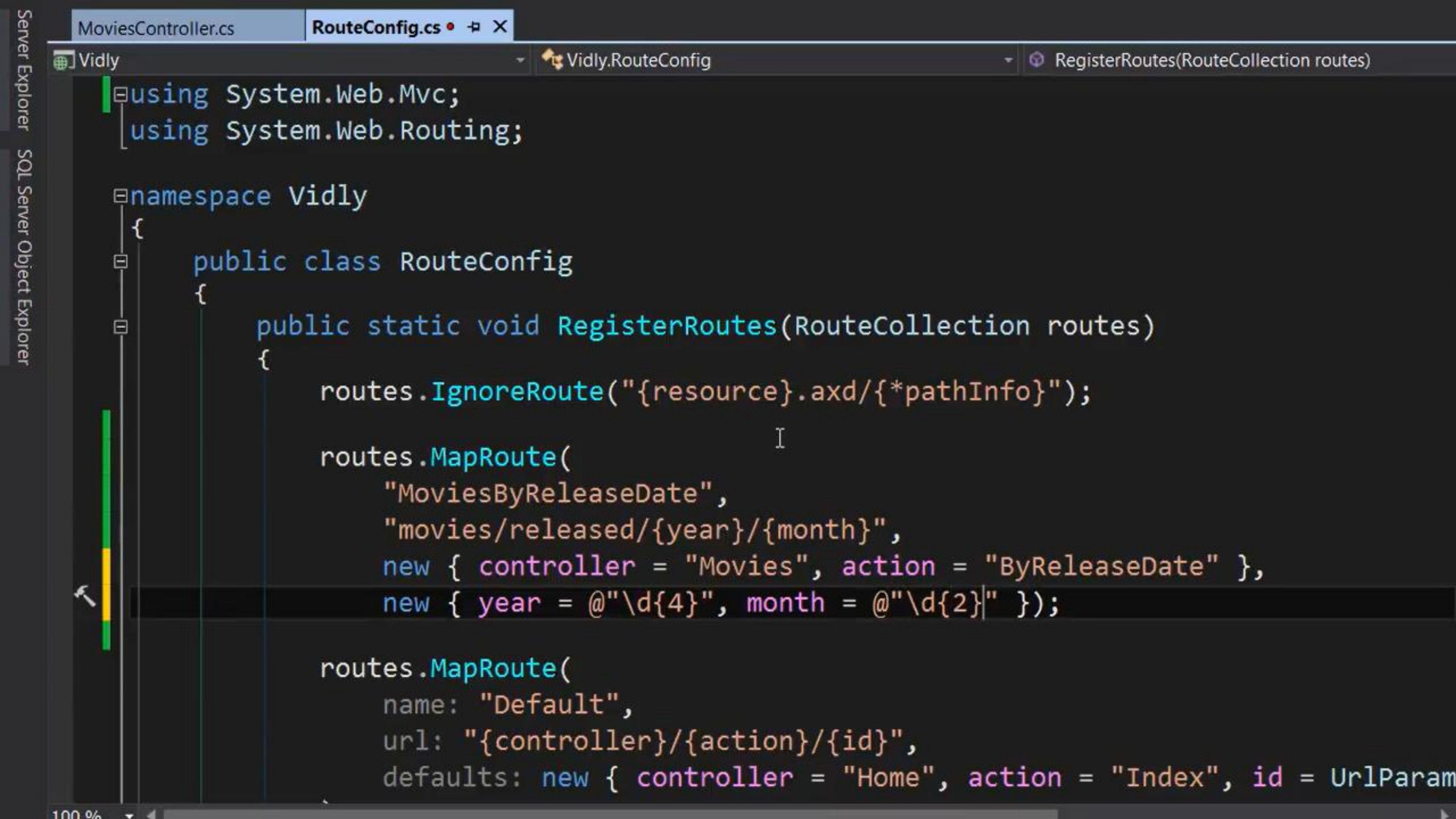

The resource cannot
Moshfegh
localhost:49333/movies/released
Apps
For quick access, place your bookmarks here on the bookmarks bar.
Import bookmarks now...

Server Error in '/' Application.

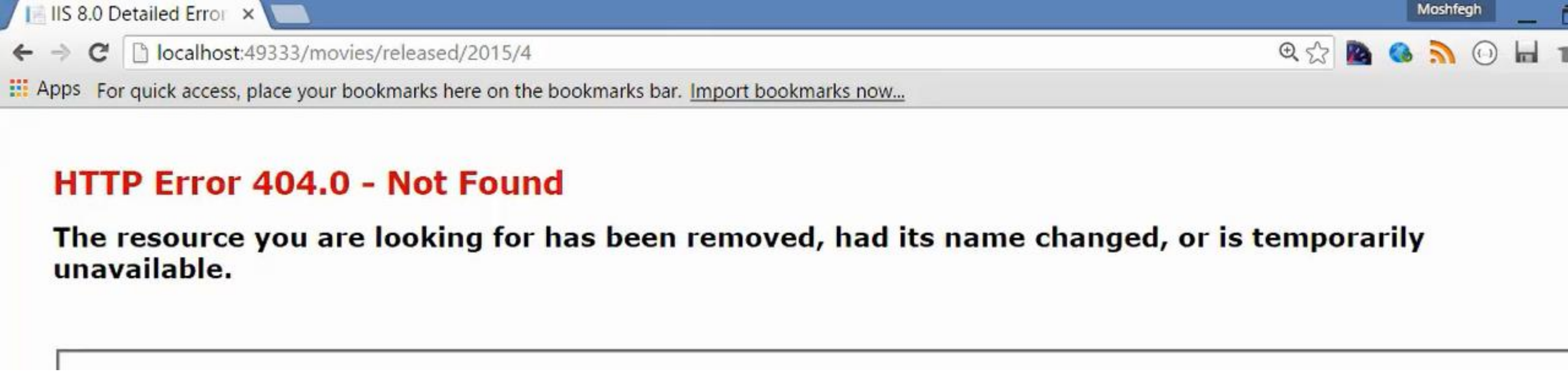
The resource cannot be found.


localhost:49333/mo
Moshfegh
localhost:49333/movies/released/2015/4
Apps
For quick access, place your bookmarks here on the bookmarks bar.
Import bookmarks now...

```
2015/4
```

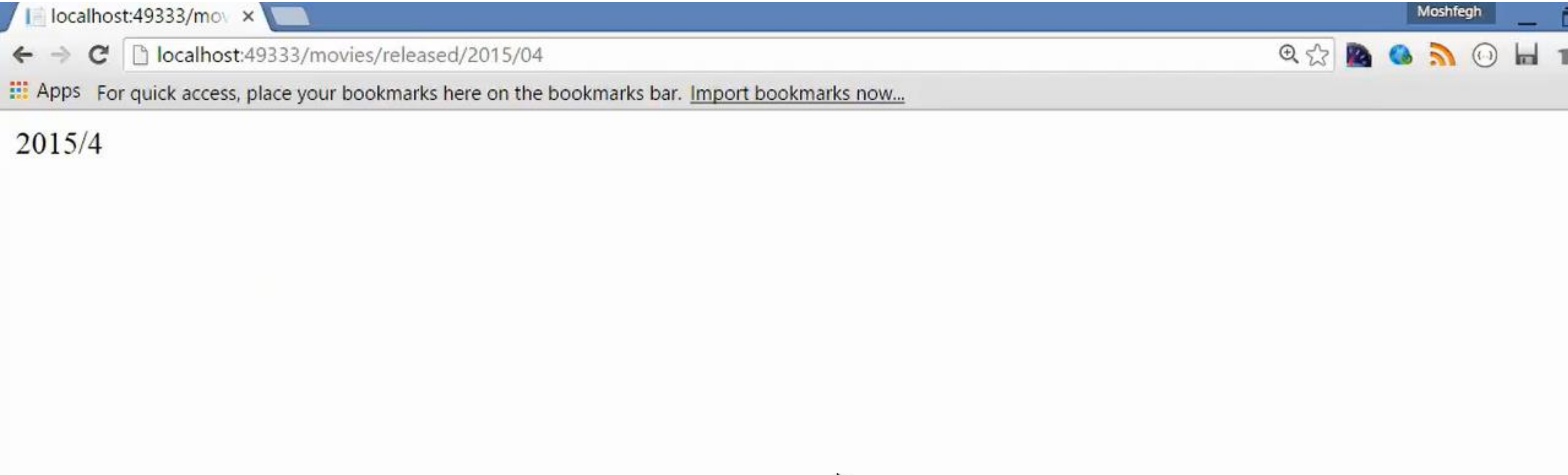


```
using System.Web.Mvc;
using System.Web.Routing;

namespace Vidly
{
    public class RouteConfig
    {
        public static void RegisterRoutes(RouteCollection routes)
        {
            routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

            routes.MapRoute(
                "MoviesByReleaseDate",
                "movies/released/{year}/{month}",
                new { controller = "Movies", action = "ByReleaseDate" },
                new { year = @"\d{4}", month = @"\d{2}" });

            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Home", action = "Index", id = UrlParam
```

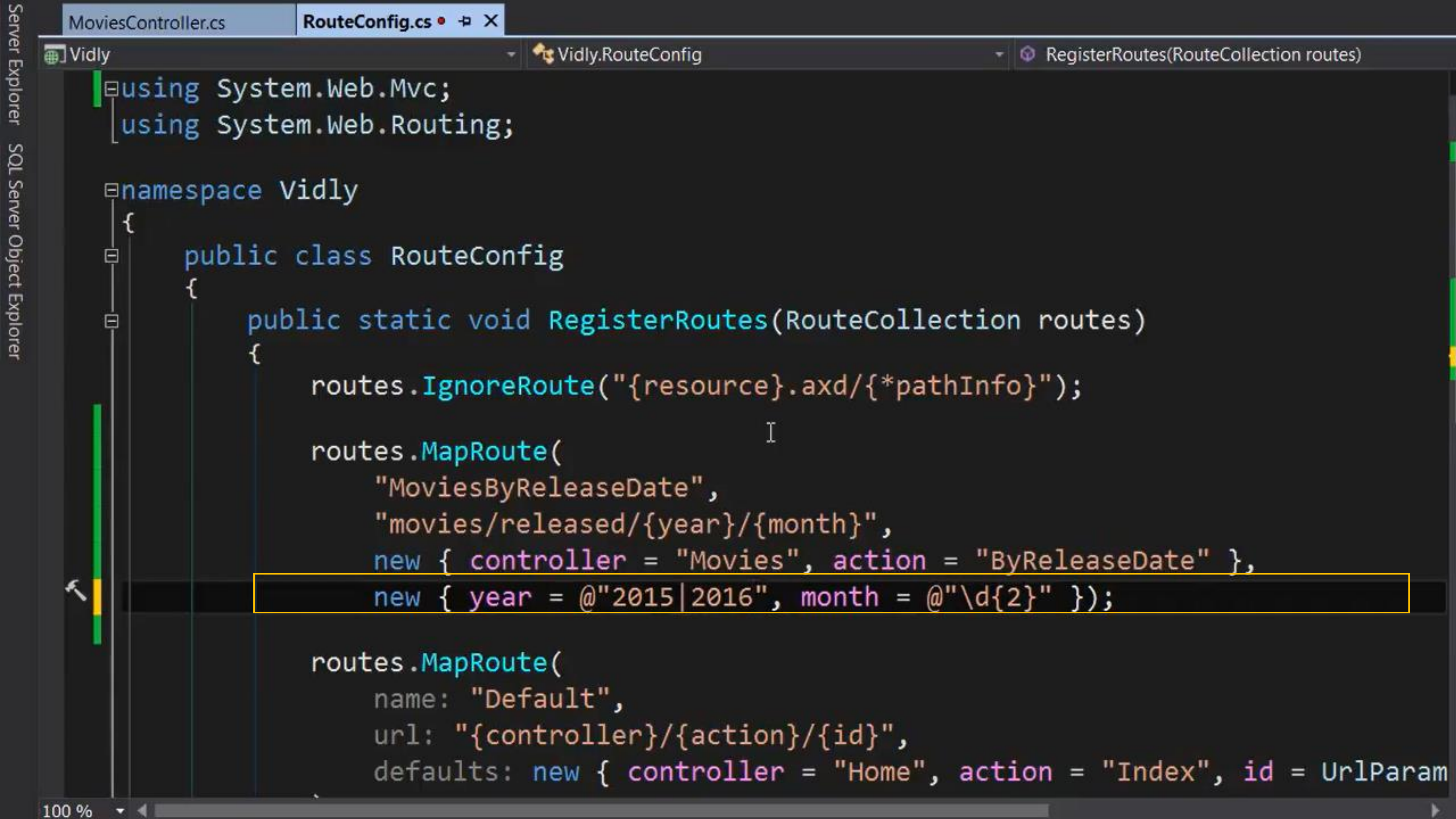

IIS 8.0 Detailed Error
Moshfegh
localhost:49333/movies/released/2015/4
Apps
For quick access, place your bookmarks here on the bookmarks bar.
Import bookmarks now...

HTTP Error 404.0 - Not Found

The resource you are looking for has been removed, had its name changed, or is temporarily unavailable.


localhost:49333/mov
Moshfegh
localhost:49333/movies/released/2015/04
Apps
For quick access, place your bookmarks here on the bookmarks bar.
Import bookmarks now...

```
2015/4
```

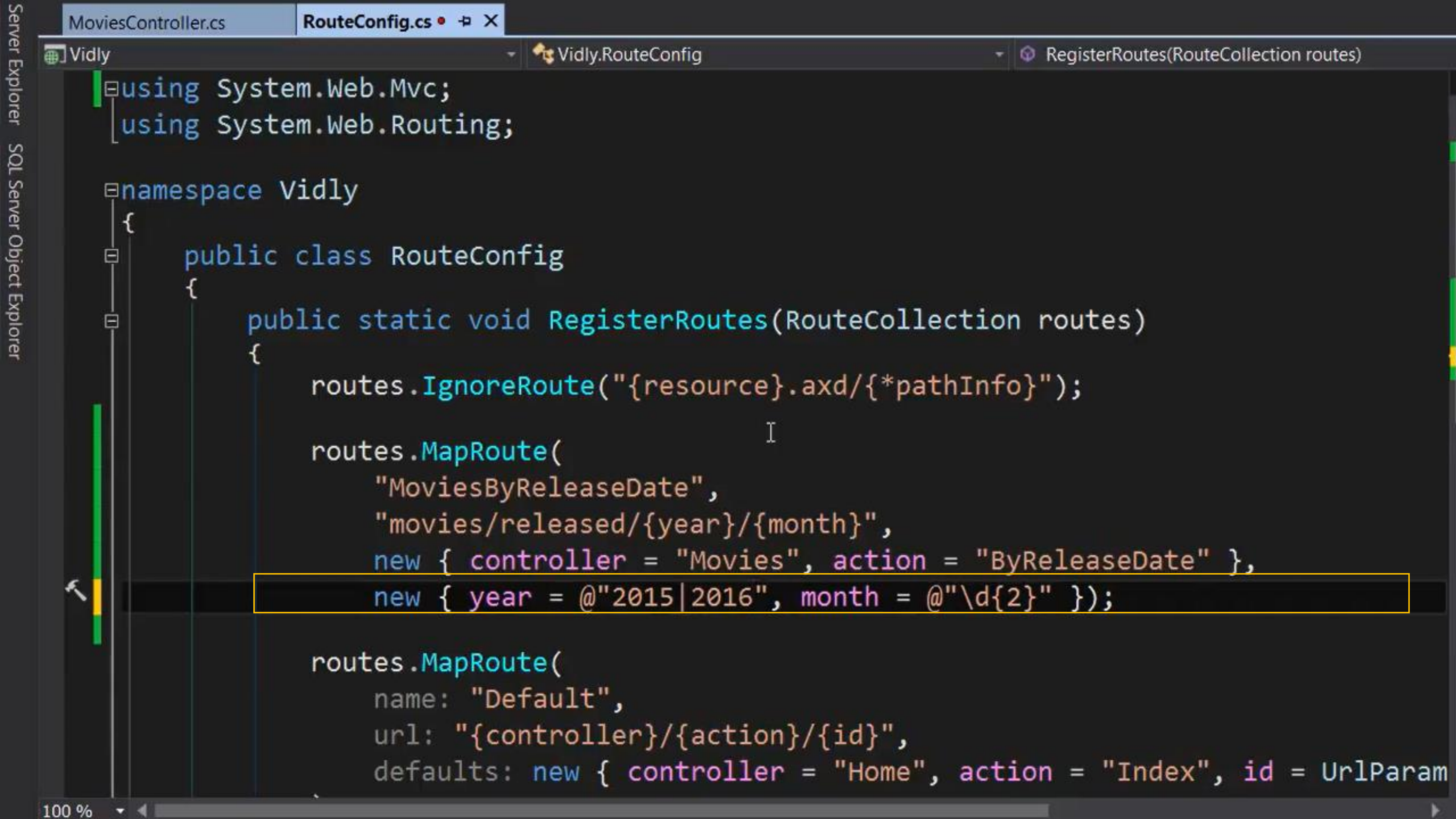


```
using System.Web.Mvc;
using System.Web.Routing;

namespace Vidly
{
    public class RouteConfig
    {
        public static void RegisterRoutes(RouteCollection routes)
        {
            routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

            routes.MapRoute(
                "MoviesByReleaseDate",
                "movies/released/{year}/{month}",
                new { controller = "Movies", action = "ByReleaseDate" },
                new { year = @"2015|2016", month = @"\d{2}" });

            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Home", action = "Index", id = UrlParam
```

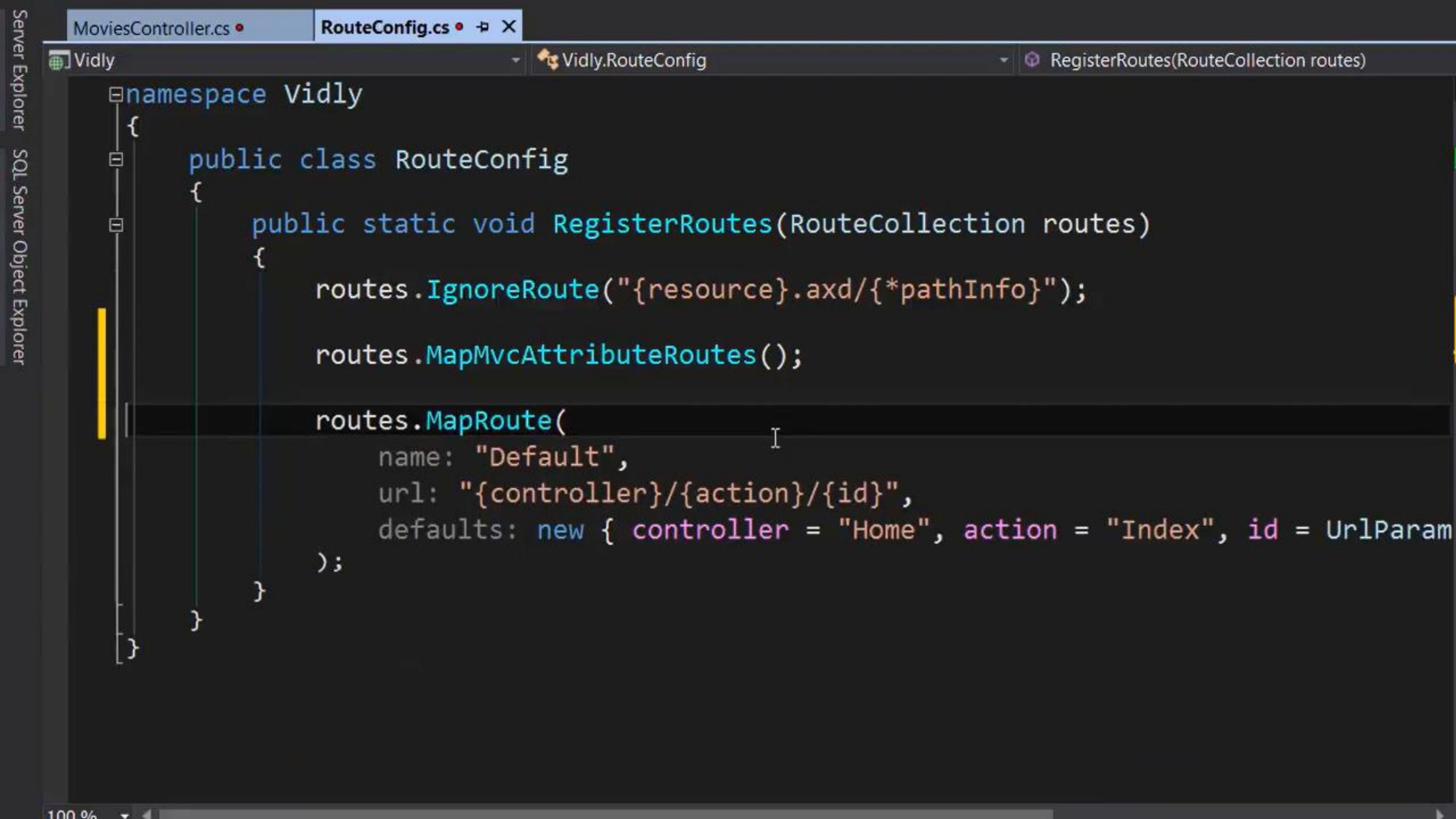
Attribute Routing

Zouhair Rimale, Ph.D.

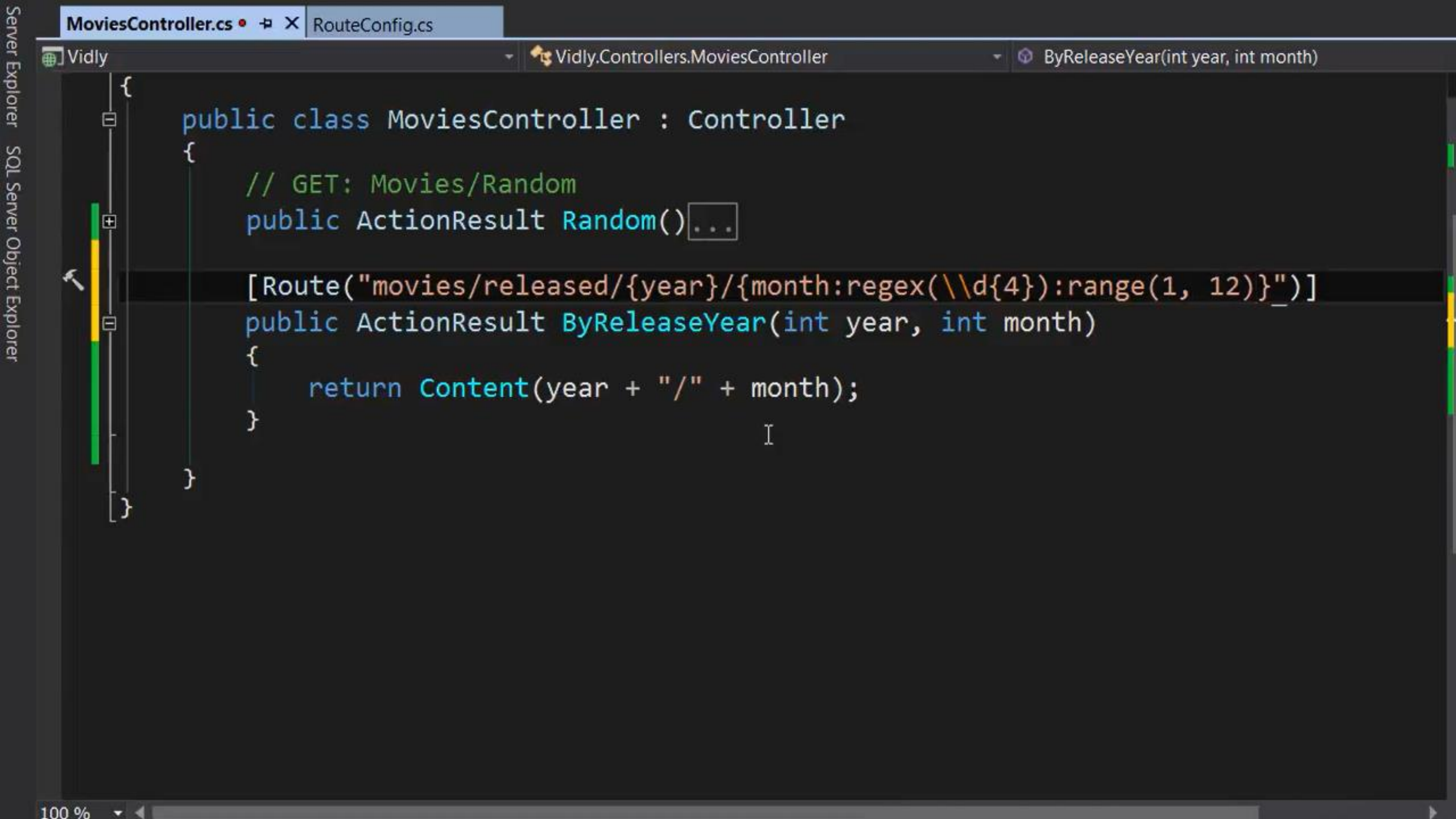
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



```
using System.Web.Mvc;
using System.Web.Routing;

namespace Vidly
{
    public class RouteConfig
    {
        public static void RegisterRoutes(RouteCollection routes)
        {
            routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

            routes.MapRoute(
                "MoviesByReleaseDate",
                "movies/released/{year}/{month}",
                new { controller = "Movies", action = "ByReleaseDate" },
                new { year = @"2015|2016", month = @"\d{2}" });

            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Home", action = "Index", id = UrlParam
```



```
namespace Vidly
{
    public class RouteConfig
    {
        public static void RegisterRoutes(RouteCollection routes)
        {
            routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

            routes.MapMvcAttributeRoutes();

            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Home", action = "Index", id = UrlParam
            );
        }
    }
}
```


Server Explorer
SQL Server Object Explorer
MoviesController.cs
RouteConfig.cs
Vidly
Vidly.Controllers.MoviesController
ByReleaseYear(int year, int month)
100 %

```
{
    public class MoviesController : Controller
    {
        // GET: Movies/Random
        public ActionResult Random()...

        [Route("movies/released/{year}/{month:regex(\\d{4}):range(1, 12)}")]
        public ActionResult ByReleaseYear(int year, int month)
        {
            return Content(year + "/" + month);
        }

    }
}
```

Constraints

- min
- max
- minlength
- maxlength
- int
- float
- guid

ASP.NET MVC Attribute Route Constraints

Controllers

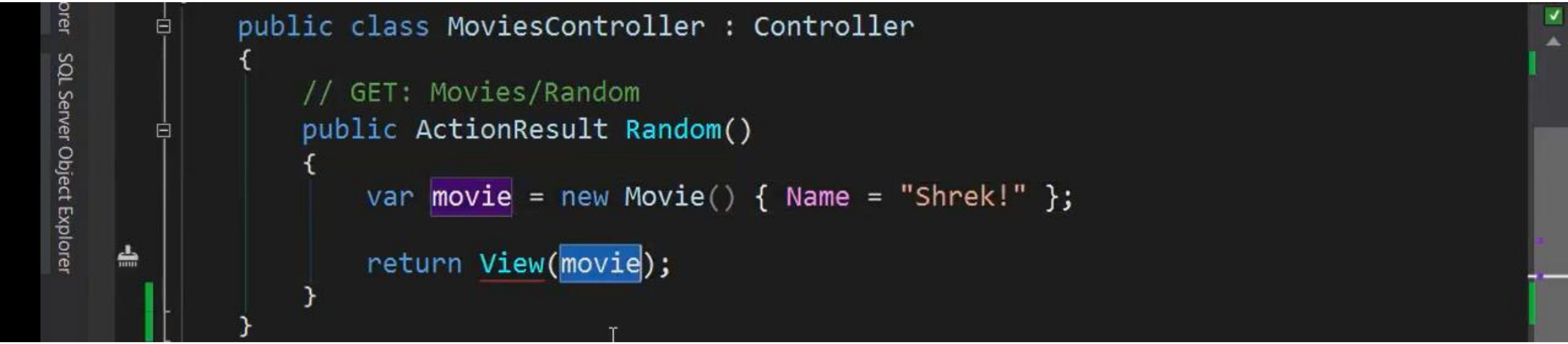
Routing

Views

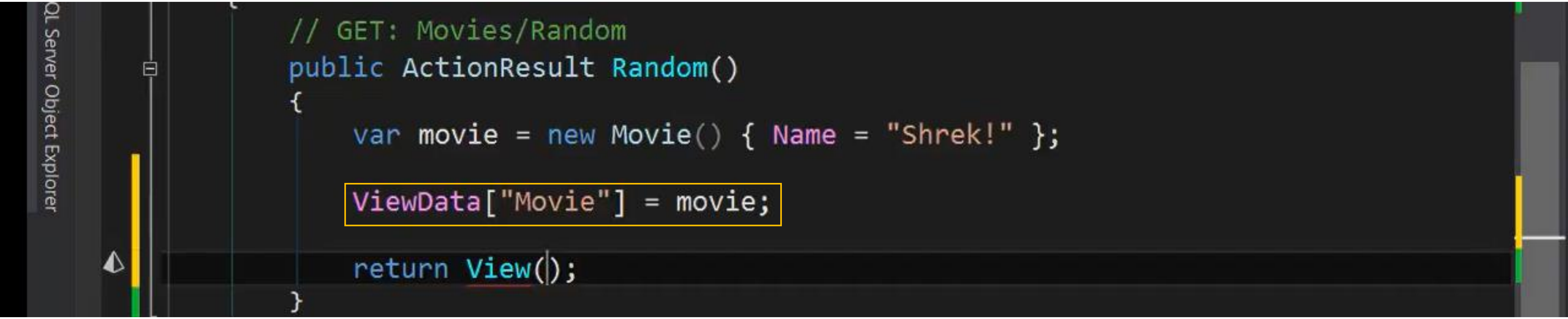
Passing Data to Views

Zouhair Rimale, Ph.D.

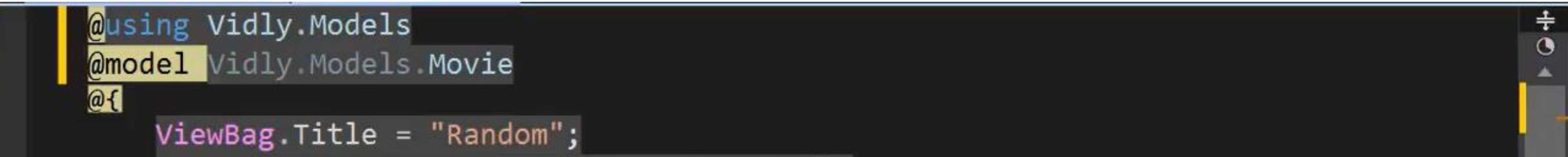
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



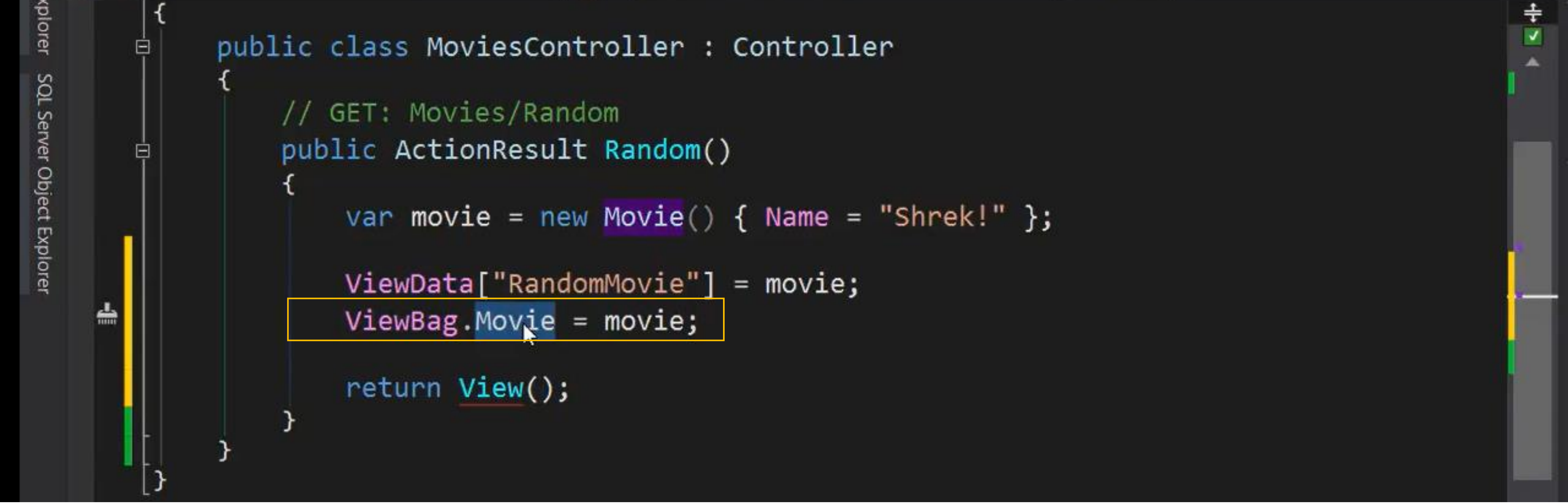
```
public class MoviesController : Controller
{
    // GET: Movies/Random
    public ActionResult Random()
    {
        var movie = new Movie() { Name = "Shrek!" };

        return View(movie);
    }
}
```



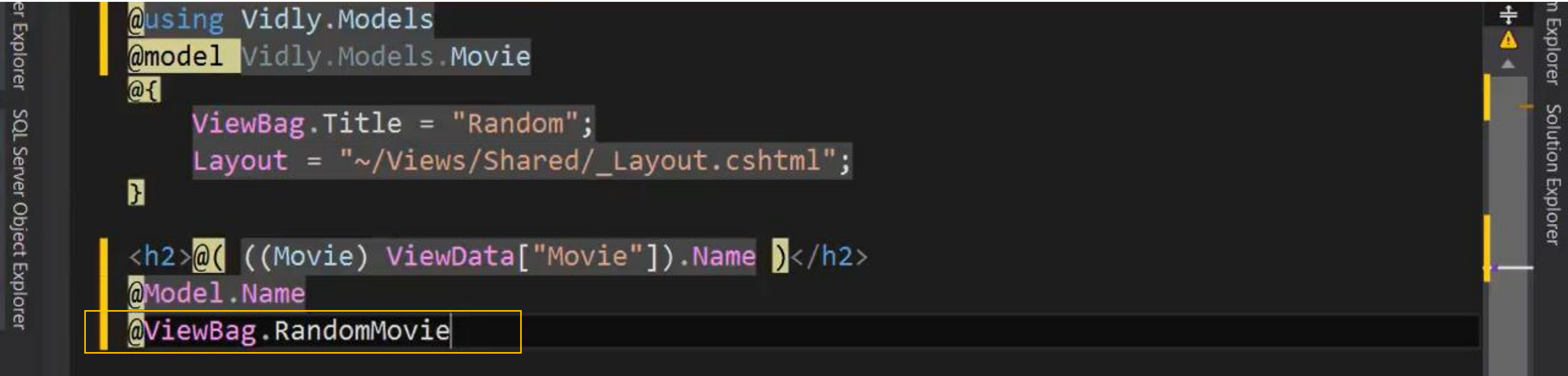
```
    // GET: Movies/Random
    public ActionResult Random()
    {
        var movie = new Movie() { Name = "Shrek!" };

        ViewData["Movie"] = movie;

        return View();
    }
```



```
@using Vidly.Models
@model Vidly.Models.Movie
@{
    ViewBag.Title = "Random";
```



```
{
    public class MoviesController : Controller
    {
        // GET: Movies/Random
        public ActionResult Random()
        {
            var movie = new Movie() { Name = "Shrek!" };

            ViewData["RandomMovie"] = movie;
            ViewBag.Movie = movie;

            return View();
        }
    }
}
```

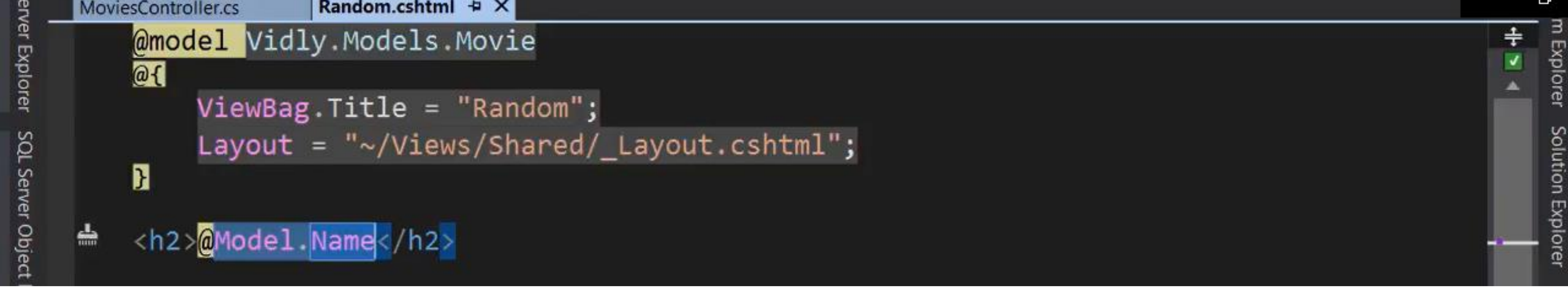


```
@using Vidly.Models
@model Vidly.Models.Movie
@{
    ViewBag.Title = "Random";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<h2>@( ((Movie) ViewData["Movie"]).Name )</h2>
@Model.Name
@ViewBag.RandomMovie
```

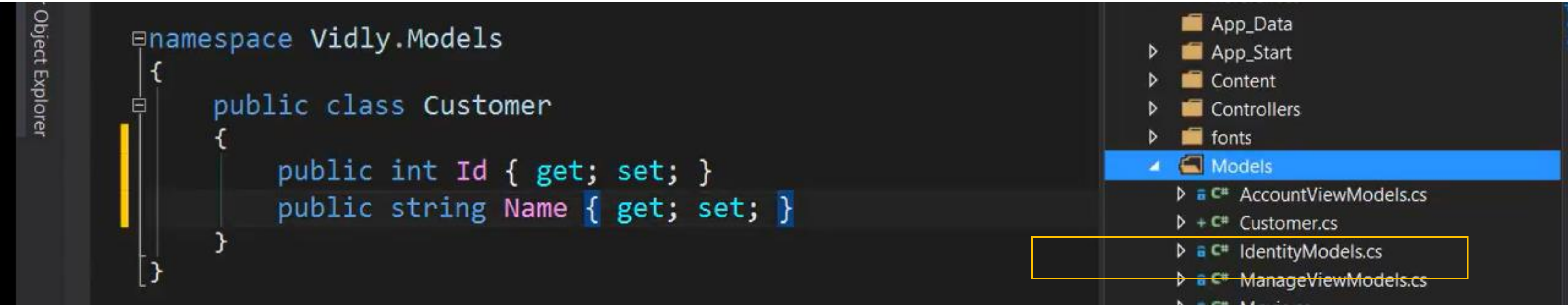
View Models

Zouhair Rimale, Ph.D.

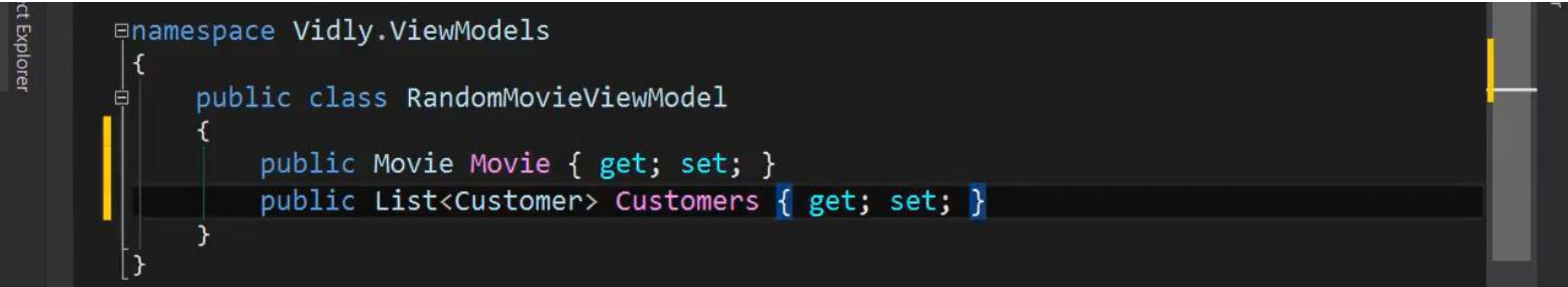
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



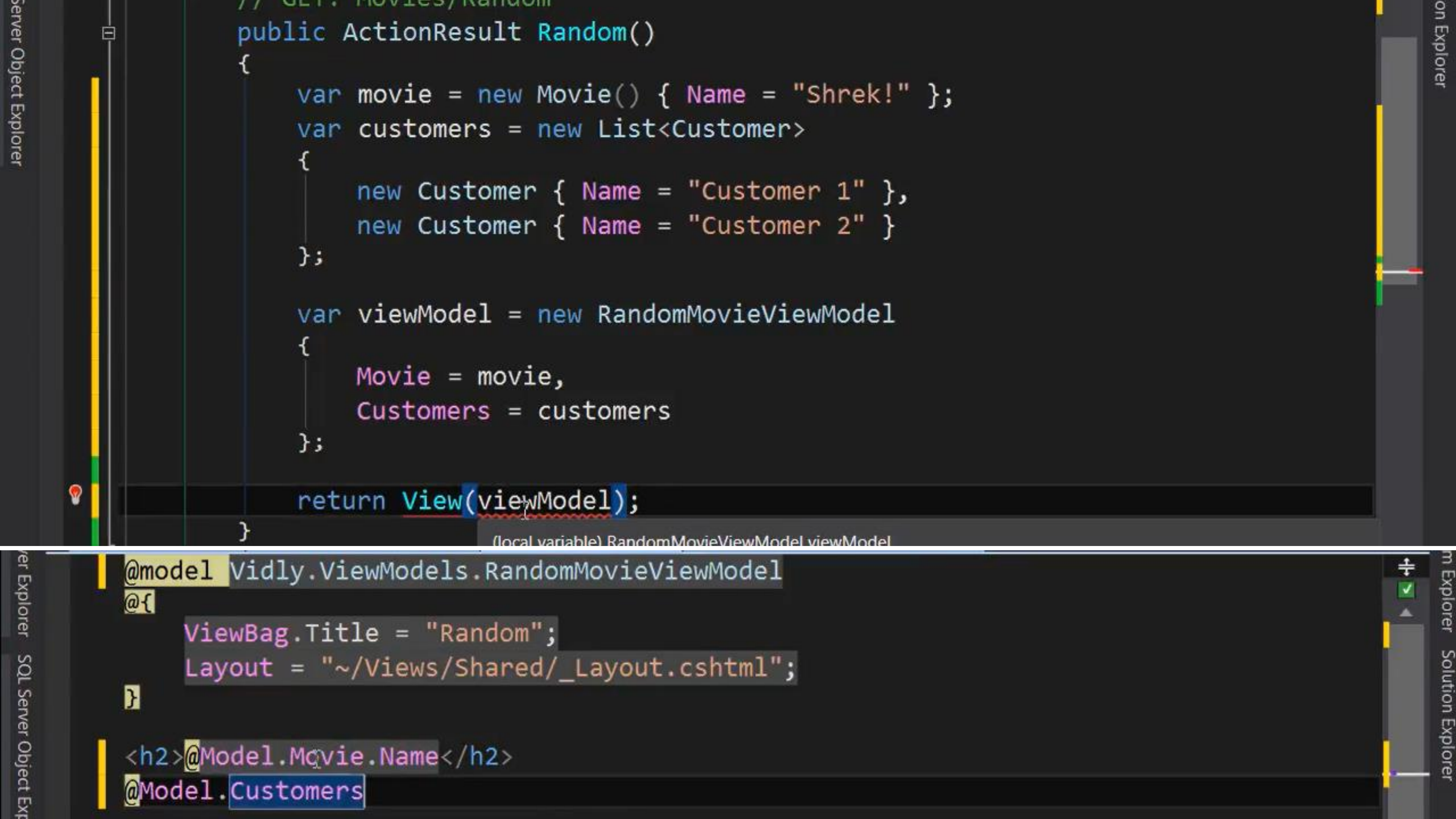
```
@model Vidly.Models.Movie
@{
    ViewBag.Title = "Random";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<h2>@Model.Name</h2>
```



```
namespace Vidly.Models
{
    public class Customer
    {
        public int Id { get; set; }
        public string Name { get; set; }
    }
}
```



```
namespace Vidly.ViewModels
{
    public class RandomMovieViewModel
    {
        public Movie Movie { get; set; }
        public List<Customer> Customers { get; set; }
    }
}
```



```
// GET: Movies/Random
public ActionResult Random()
{
    var movie = new Movie() { Name = "Shrek!" };
    var customers = new List<Customer>
    {
        new Customer { Name = "Customer 1" },
        new Customer { Name = "Customer 2" }
    };

    var viewModel = new RandomMovieViewModel
    {
        Movie = movie,
        Customers = customers
    };

    return View(viewModel);
}
```

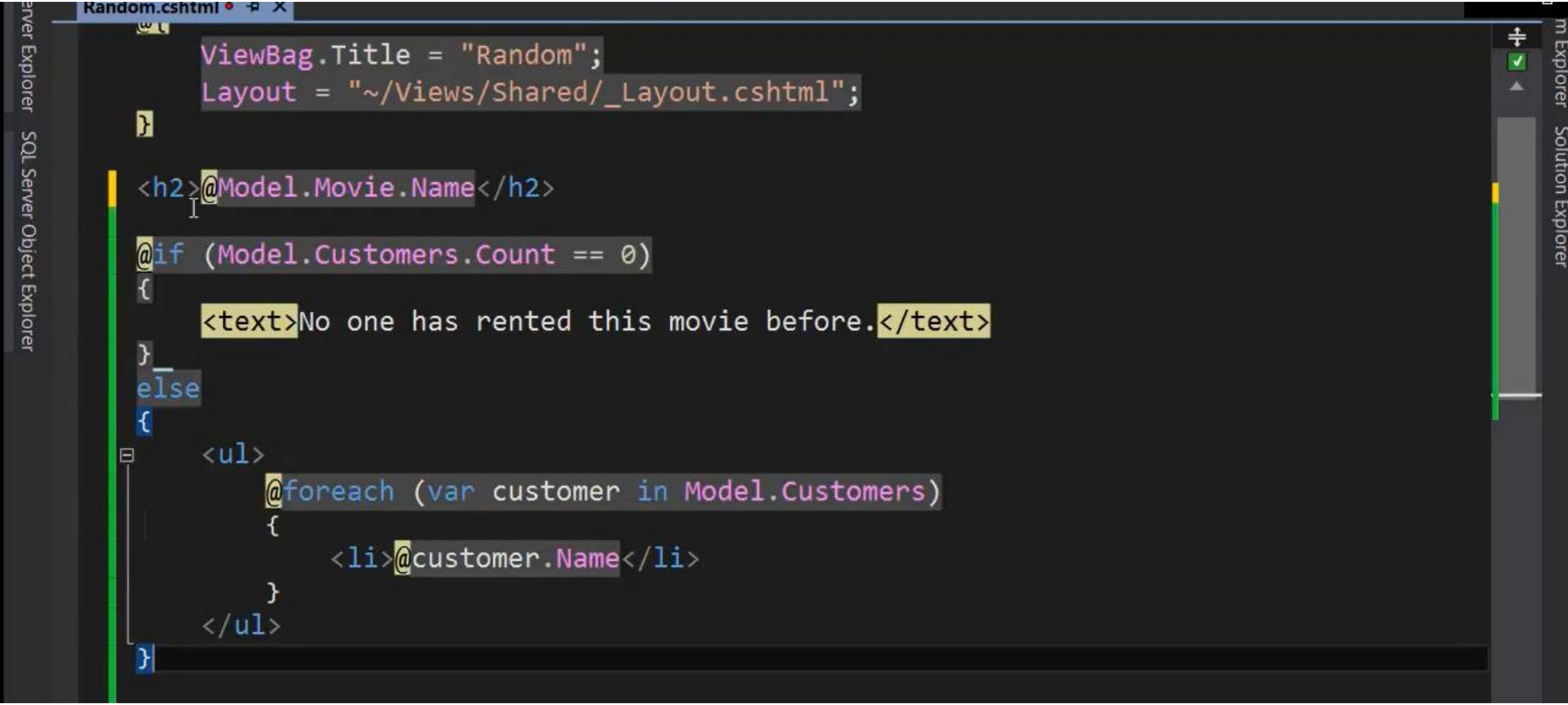
(local variable) RandomMovieViewModel viewModel

```
@model Vidly.ViewModels.RandomMovieViewModel
@{
    ViewBag.Title = "Random";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<h2>@Model.Movie.Name</h2>
@Model.Customers
```

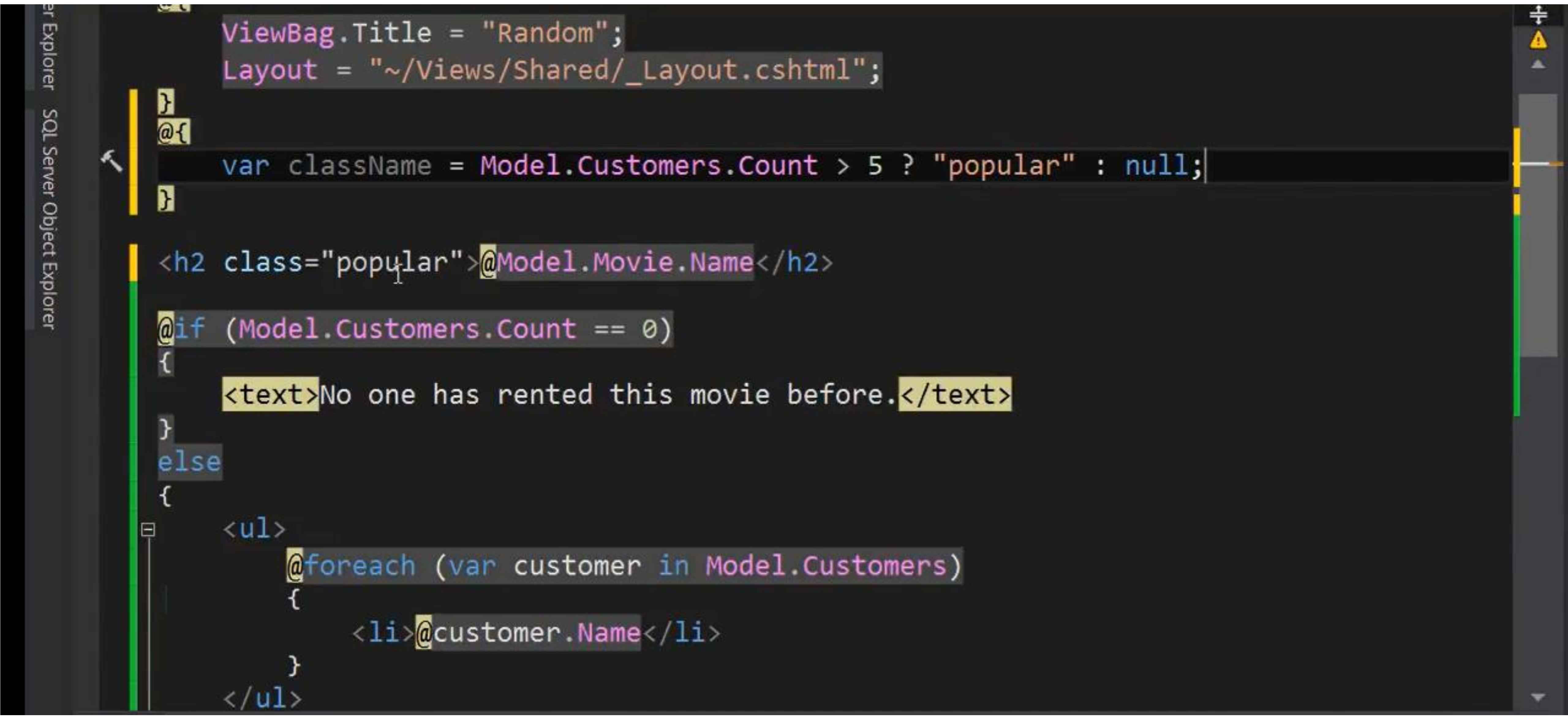
Razor View

Zouhair Rimale, Ph.D.

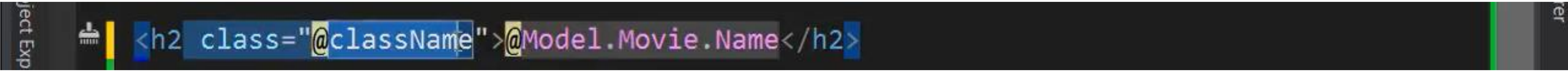
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



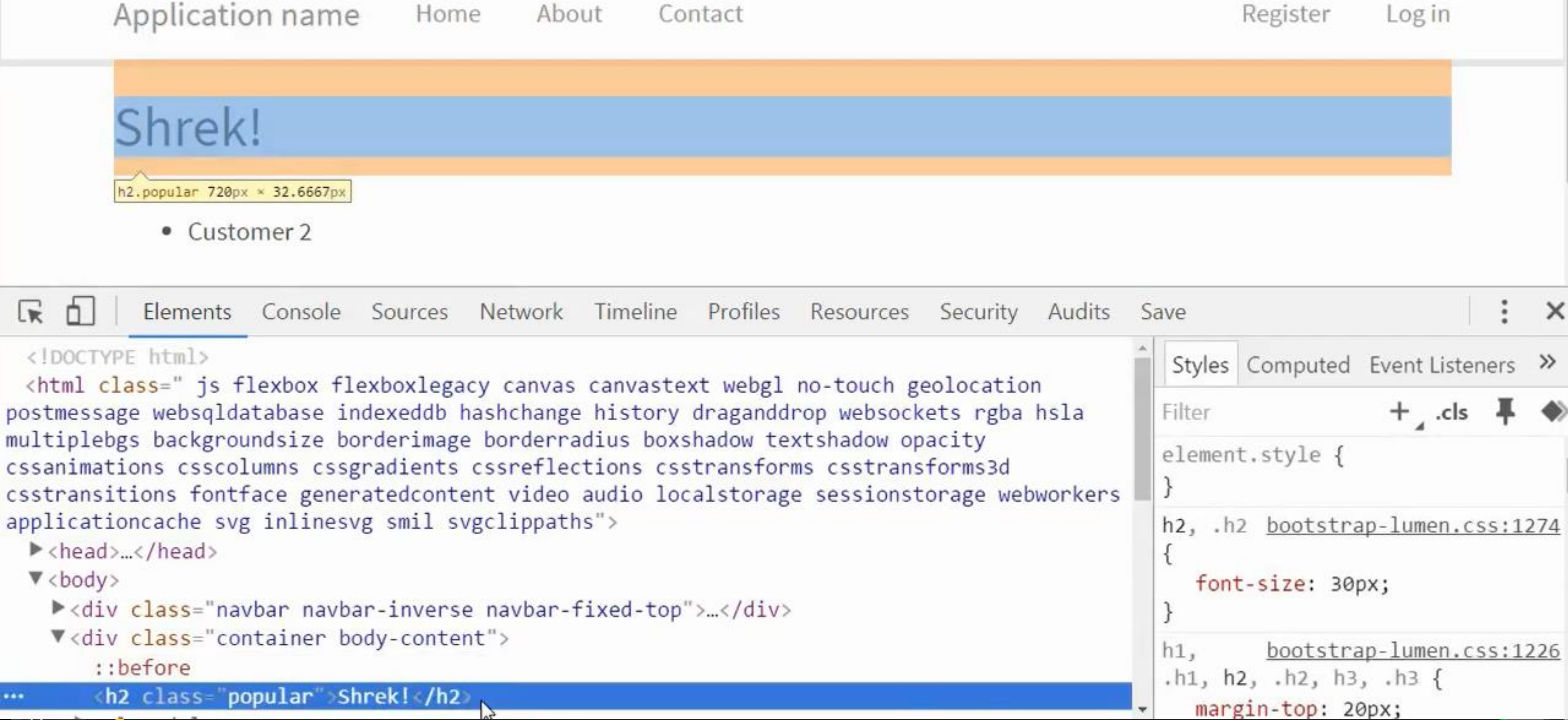
```
@{
    ViewBag.Title = "Random";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<h2>@Model.Movie.Name</h2>

@if (Model.Customers.Count == 0)
{
    <text>No one has rented this movie before.</text>
}
else
{
    <ul>
        @foreach (var customer in Model.Customers)
        {
            <li>@customer.Name</li>
        }
    </ul>
}
```



```
    ViewBag.Title = "Random";
    Layout = "~/Views/Shared/_Layout.cshtml";
}
@{
    var className = Model.Customers.Count > 5 ? "popular" : null;
}

<h2 class="popular">@Model.Movie.Name</h2>

@if (Model.Customers.Count == 0)
{
    <text>No one has rented this movie before.</text>
}
else
{
    <ul>
        @foreach (var customer in Model.Customers)
        {
            <li>@customer.Name</li>
        }
    </ul>
```



```
<h2 class="@className">@Model.Movie.Name</h2>
```



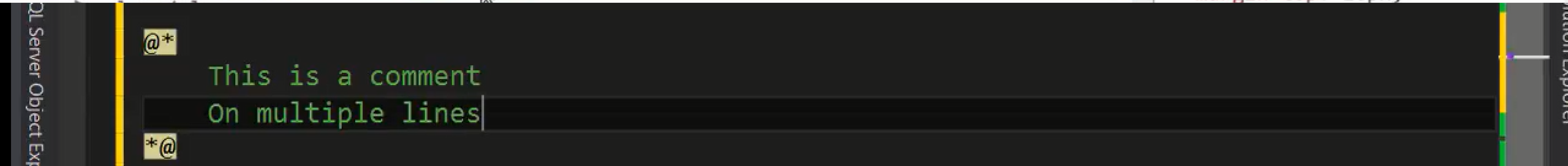
Shrek!

h2.popular 720px × 32.6667px

- Customer 2

```
<!DOCTYPE html>
<html class=" js flexbox flexboxlegacy canvas canvastext webgl no-touch geolocation
postmessage websqldatabase indexeddb hashchange history draganddrop websockets rgba hsla
multiplebgs backgroundsize borderimage borderradius boxshadow textshadow opacity
cssanimations csscolumns cssgradients cssreflections csstransforms csstransforms3d
csstransitions fontface generatedcontent video audio localstorage sessionstorage webworkers
applicationcache svg inlinesvg smil svgclippaths">
  ▶<head>…</head>
  ▼<body>
    ▶<div class="navbar navbar-inverse navbar-fixed-top">…</div>
    ▼<div class="container body-content">
        ::before
        <h2 class="popular">Shrek!</h2>
```

```
element.style {
}
h2, .h2 bootstrap-lumen.css:1274
{
  font-size: 30px;
}
h1,        bootstrap-lumen.css:1226
.h1, h2, .h2, h3, .h3 {
  margin-top: 20px;
```

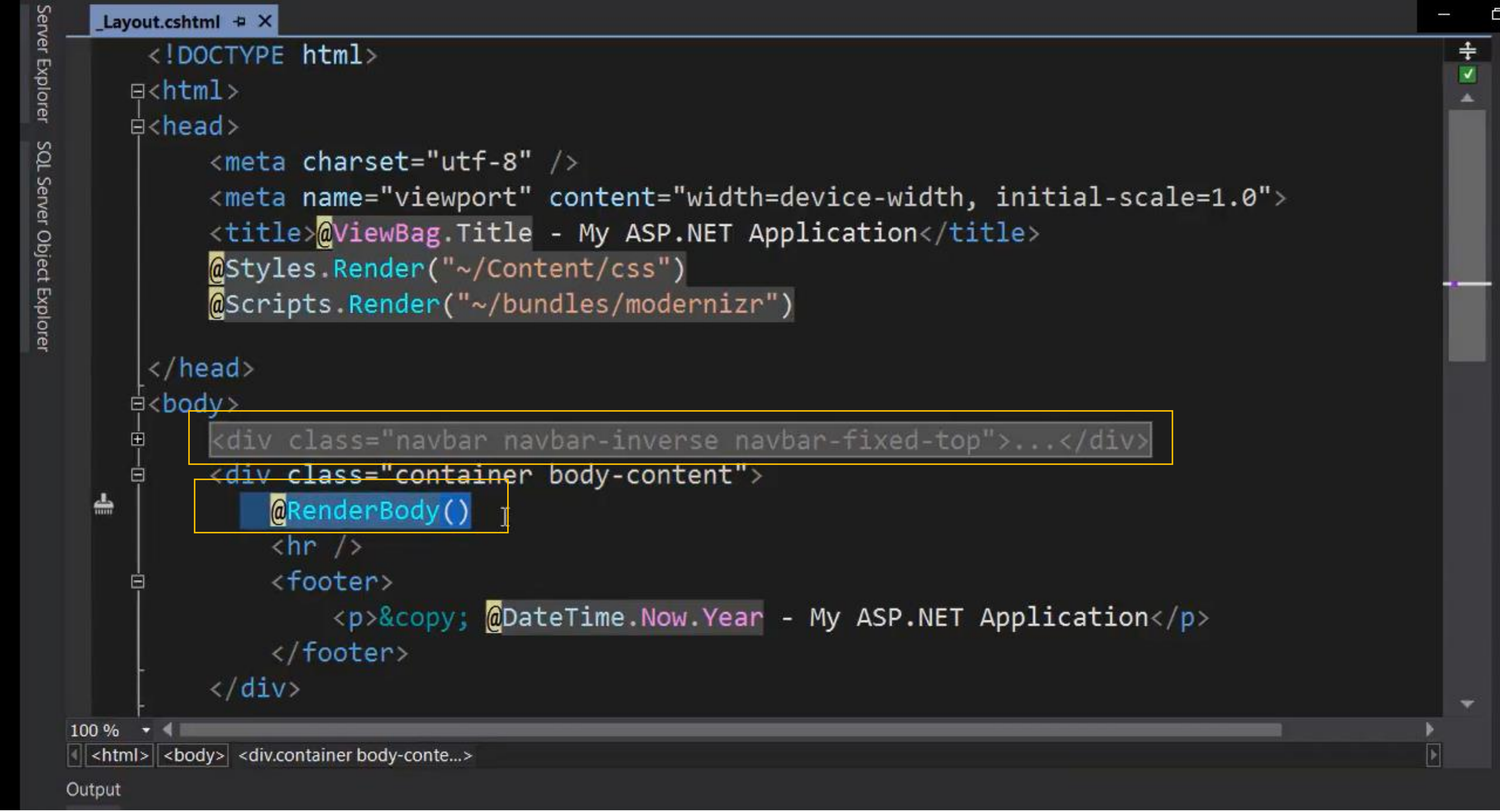


```
@*
    This is a comment
    On multiple lines
*@
```

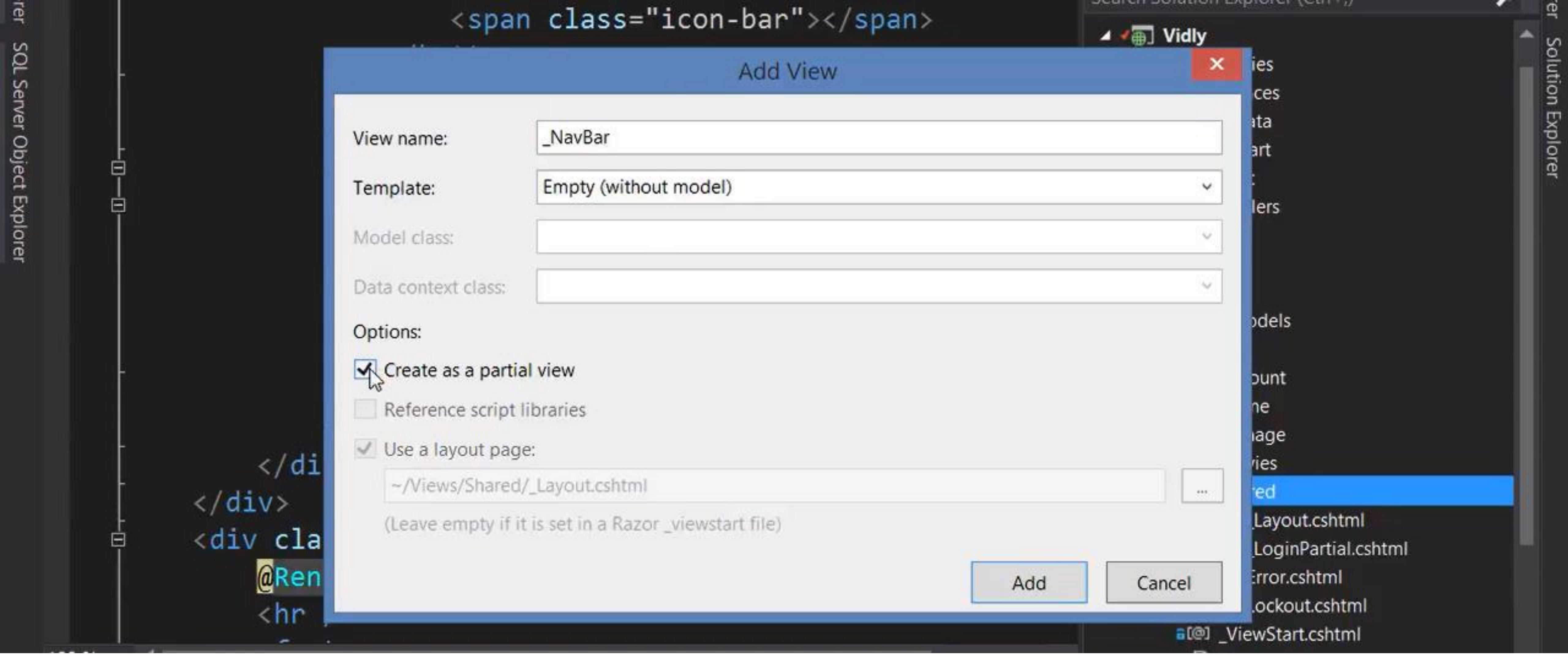
Partial Views

Zouhair Rimale, Ph.D.

Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>@ViewBag.Title - My ASP.NET Application</title>
    @Styles.Render("~/Content/css")
    @Scripts.Render("~/bundles/modernizr")

</head>
<body>
    <div class="navbar navbar-inverse navbar-fixed-top">...</div>
    <div class="container body-content">
        @RenderBody()
        <hr />
        <footer>
            <p>&copy; @DateTime.Now.Year - My ASP.NET Application</p>
        </footer>
    </div>
```



Add View

View name: _NavBar

Template: Empty (without model)

Model class:

Data context class:

Options:

☑ Create as a partial view

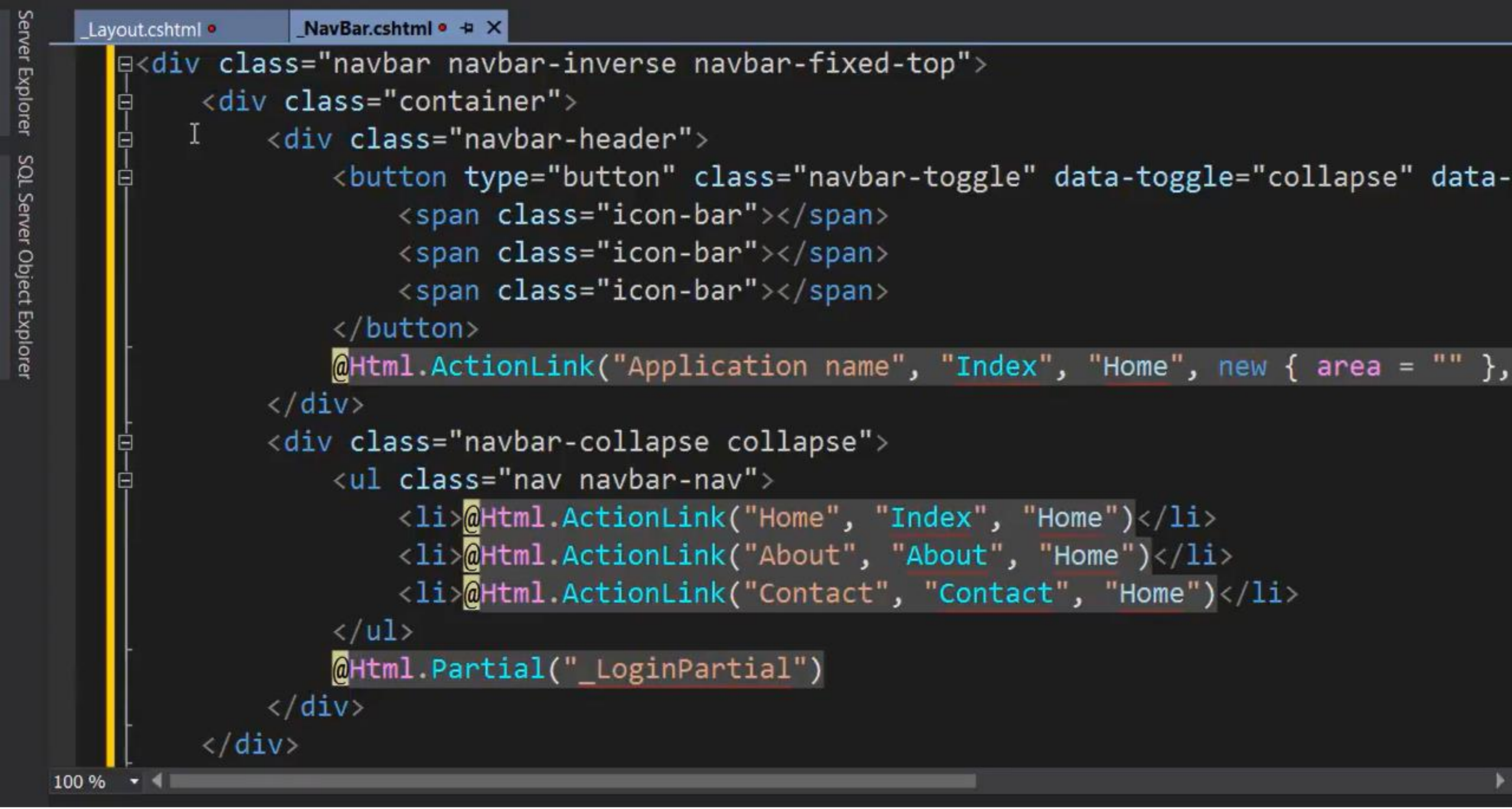
□ Reference script libraries

☑ Use a layout page:

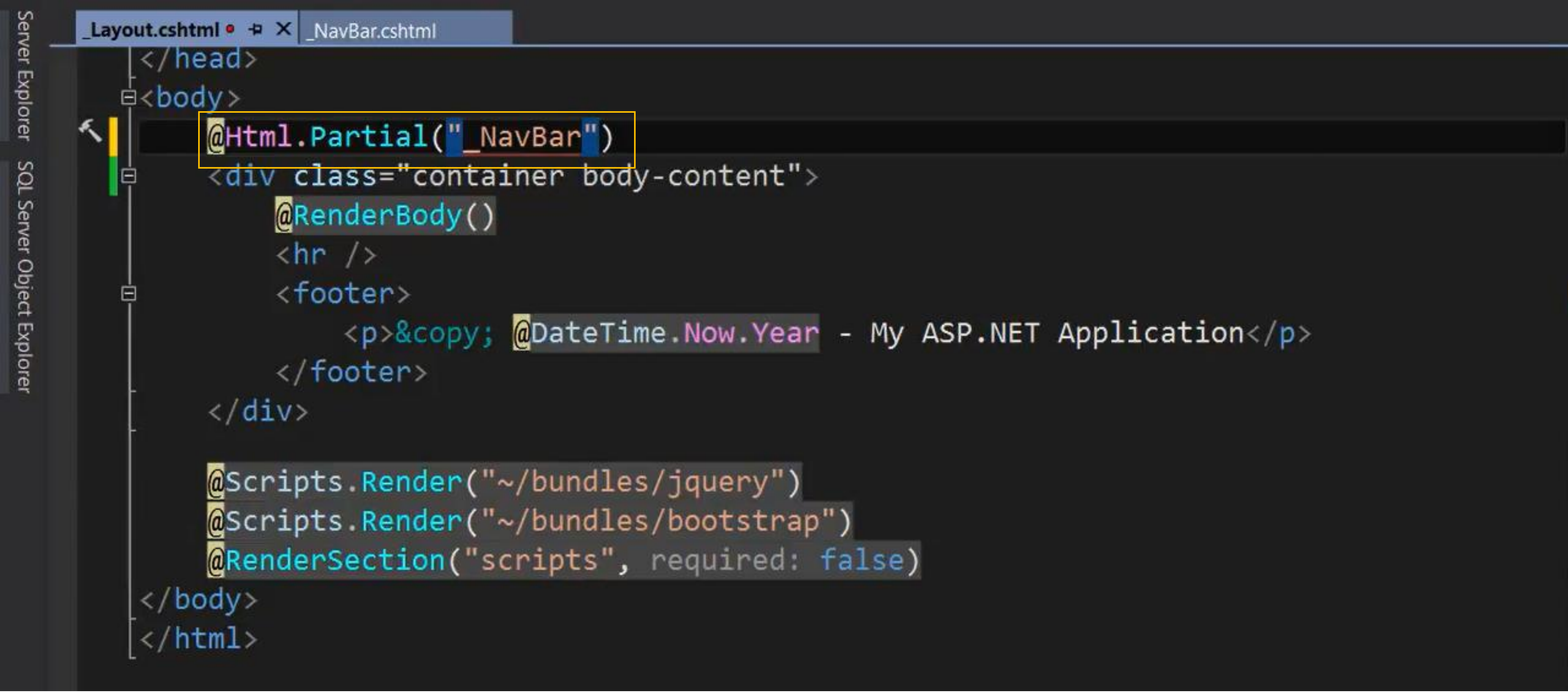
~/Views/Shared/_Layout.cshtml

(Leave empty if it is set in a Razor _viewstart file)

```
<span class="icon-bar"></span>
```

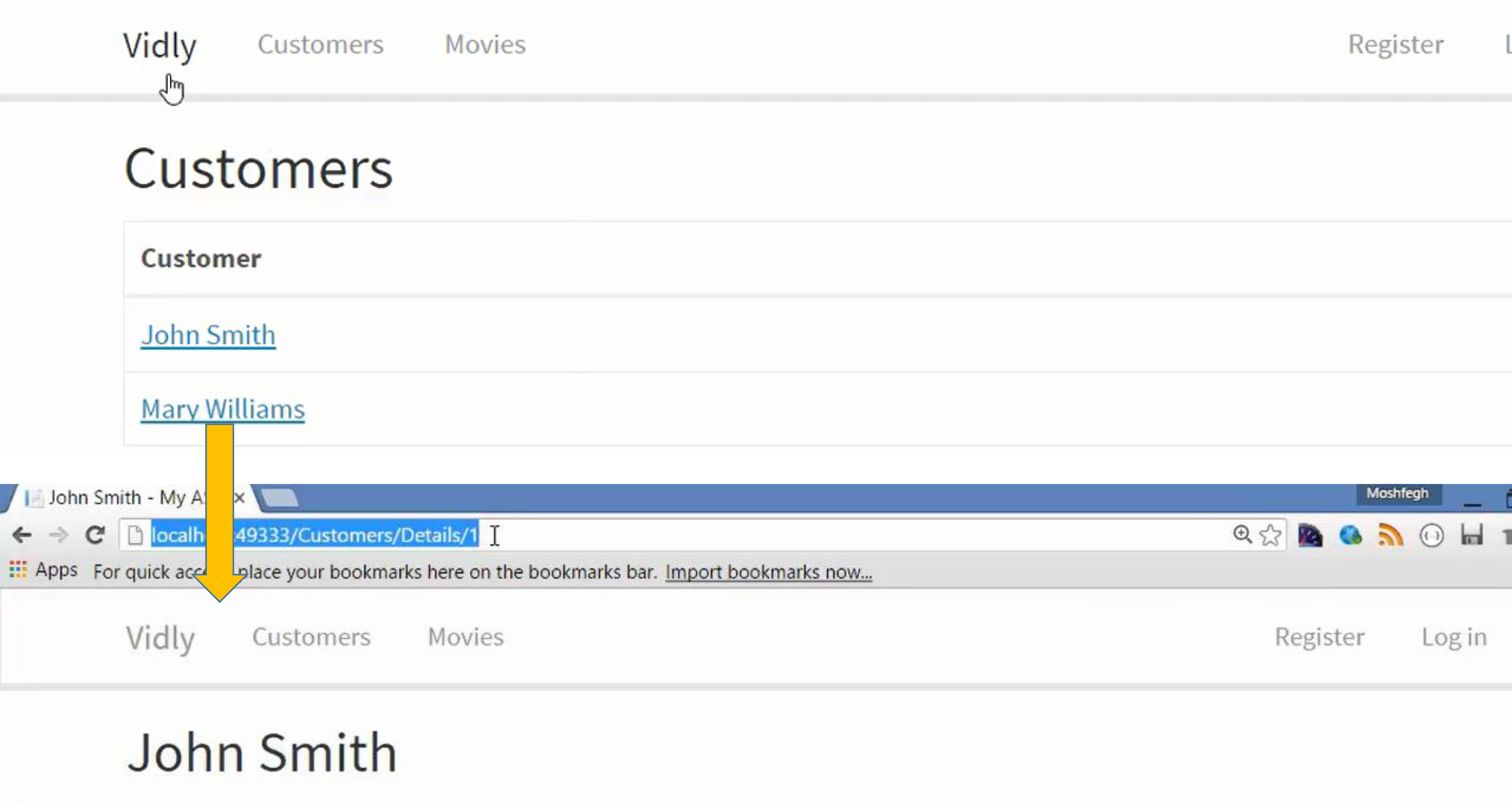


```
<div class="navbar navbar-inverse navbar-fixed-top">
    <div class="container">
        <div class="navbar-header">
            <button type="button" class="navbar-toggle" data-toggle="collapse" data-
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </button>
            @Html.ActionLink("Application name", "Index", "Home", new { area = "" },
        </div>
        <div class="navbar-collapse collapse">
            <ul class="nav navbar-nav">
                <li>@Html.ActionLink("Home", "Index", "Home")</li>
                <li>@Html.ActionLink("About", "About", "Home")</li>
                <li>@Html.ActionLink("Contact", "Contact", "Home")</li>
            </ul>
            @Html.Partial("_LoginPartial")
        </div>
    </div>
```



```
</head>
<body>
    @Html.Partial("_NavBar")
    <div class="container body-content">
        @RenderBody()
        <hr />
        <footer>
            <p>&copy; @DateTime.Now.Year - My ASP.NET Application</p>
        </footer>
    </div>

    @Scripts.Render("~/bundles/jquery")
    @Scripts.Render("~/bundles/bootstrap")
    @RenderSection("scripts", required: false)
</body>
</html>
```

Exercise

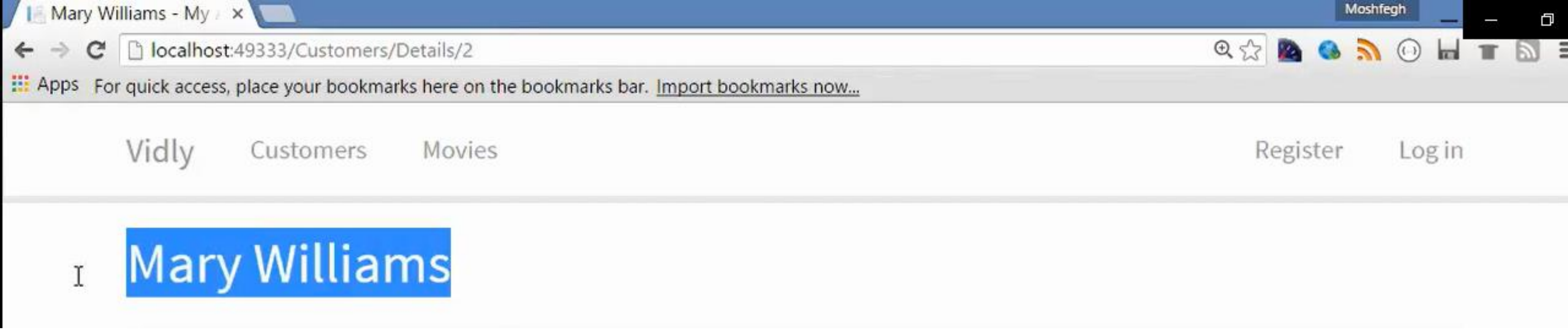

Vidly
Customers
Movies
Register

Customers

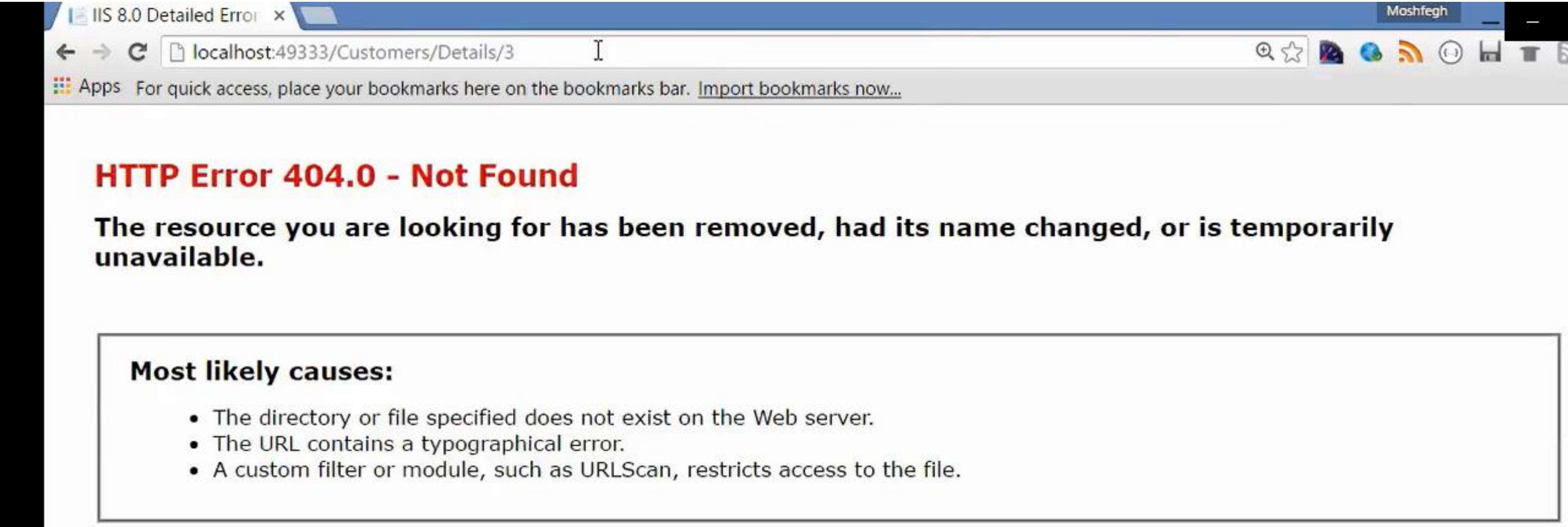
Customer
John Smith
Mary Williams

John Smith - My A
Moshfegh
localh 49333/Customers/Details/1
Apps
For quick acc place your bookmarks here on the bookmarks bar. Import bookmarks now...
Vidly
Customers
Movies
Register
Log in

John Smith


Mary Williams - My
Moshfegh
localhost:49333/Customers/Details/2
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Vidly
Customers
Movies
Register
Log in

Mary Williams

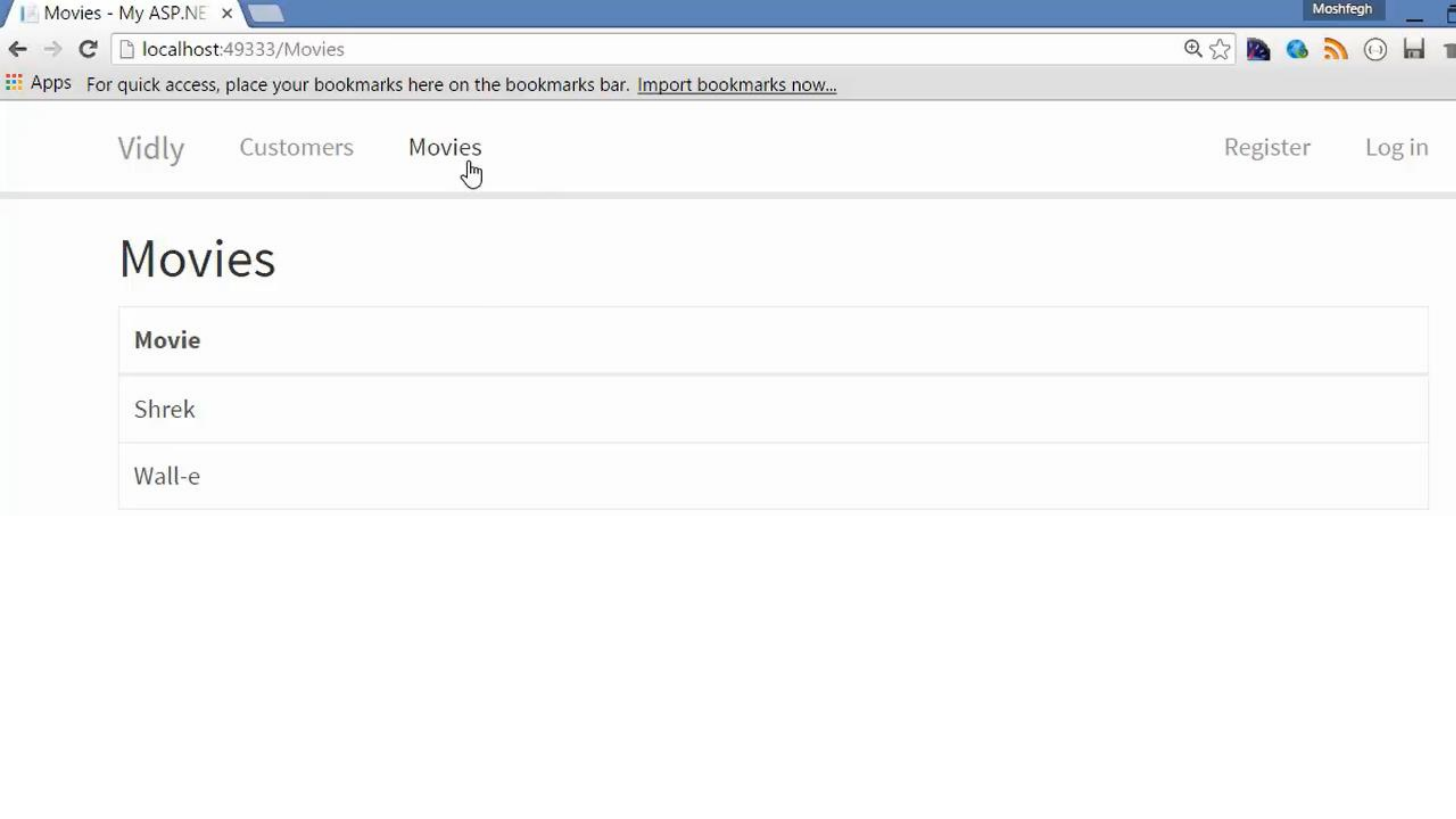

IIS 8.0 Detailed Error
Moshfegh
localhost:49333/Customers/Details/3
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

HTTP Error 404.0 - Not Found

The resource you are looking for has been removed, had its name changed, or is temporarily unavailable.

Most likely causes:

- The directory or file specified does not exist on the Web server.
- The URL contains a typographical error.
- A custom filter or module, such as URLScan, restricts access to the file.


Movies - My ASP.NE
Moshfegh
localhost:49333/Movies
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Vidly
Customers
Movies
Register
Log in

Movies

Movie
Shrek
Wall-e

Summary

ASP.NET MVC Fundamentals

Summary

- Action Results
- Action Parameters
- Convention-based Routing
- Attribute Routing
- Razor Views

Next

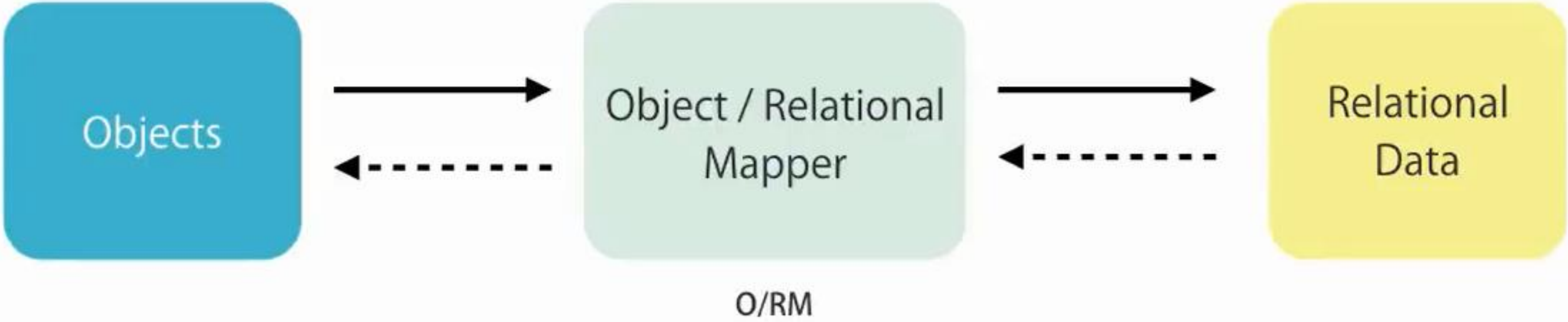
Working with Data

Entity Framework

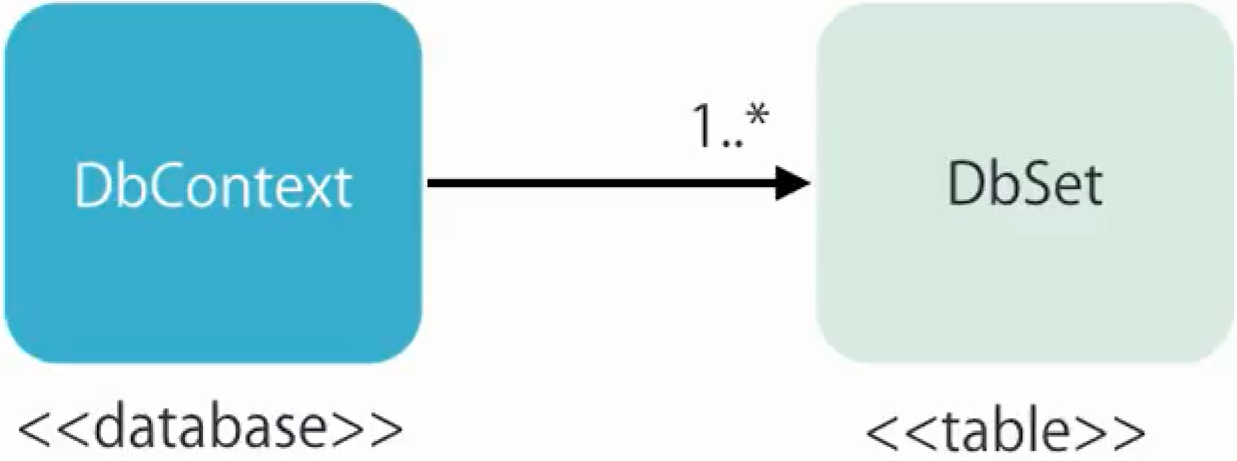
Zouhair Rimale, Ph.D.

Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular

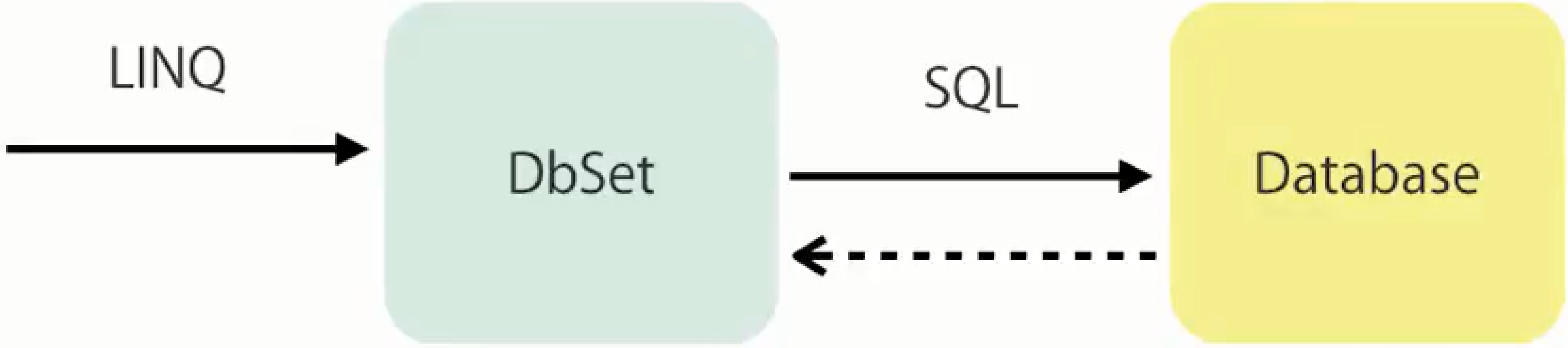
Entity Framework



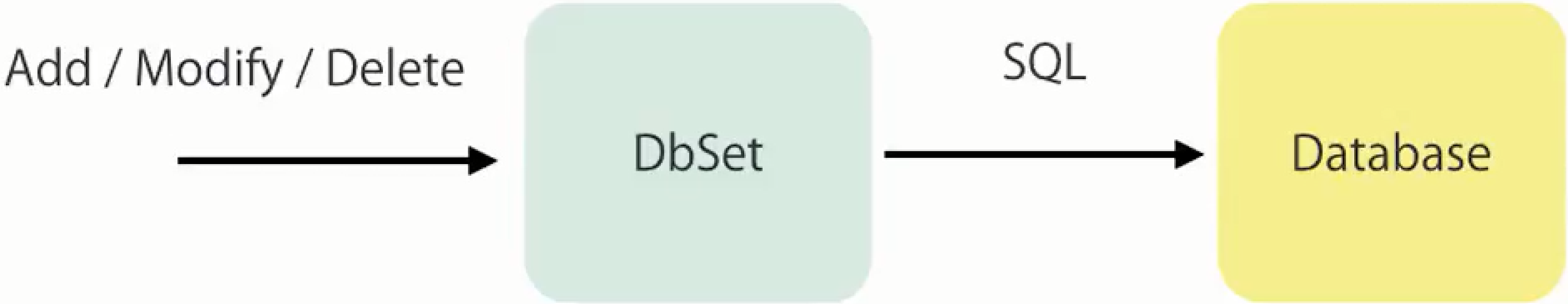
Entity Framework



LINQ



LINQ



Add / Modify / Delete

DbSet

SQL

Database

Workflows

Database First

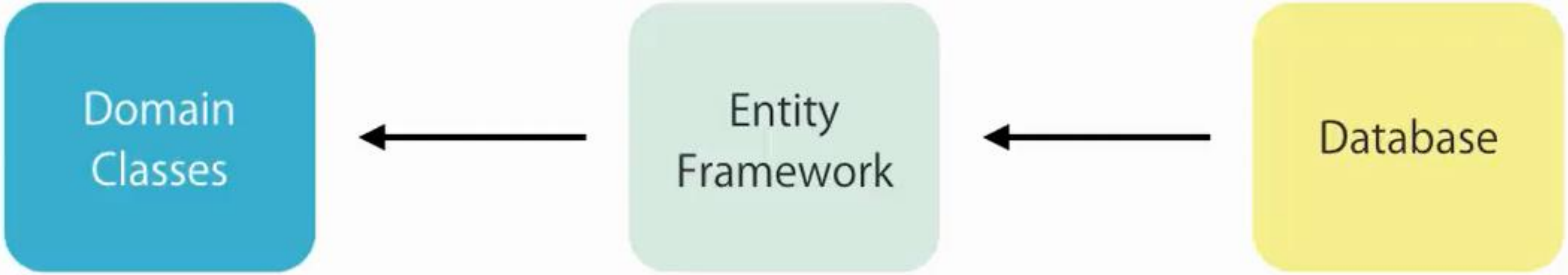
Code First

DbFirst vs CodeFirst

Zouhair Rimale, Ph.D.

Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular

DbFirst Workflow

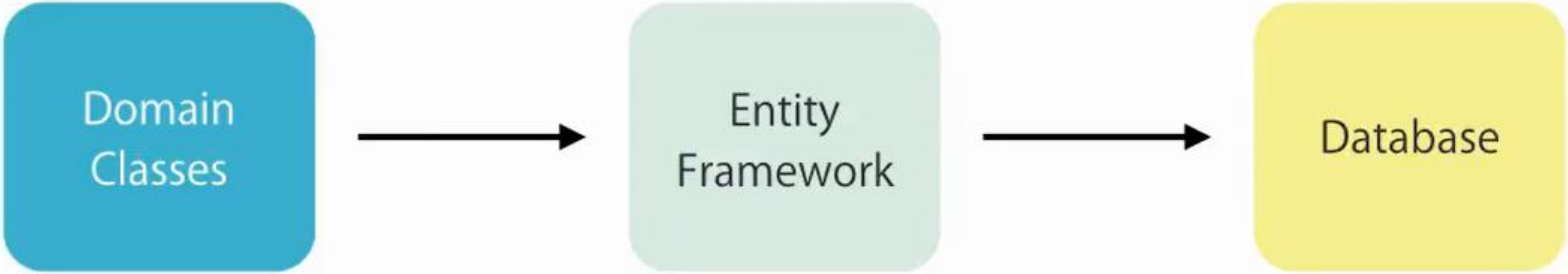


Domain Classes

Entity Framework

Database

CodeFirst Workflow



Domain Classes

Entity Framework

Database

Code First

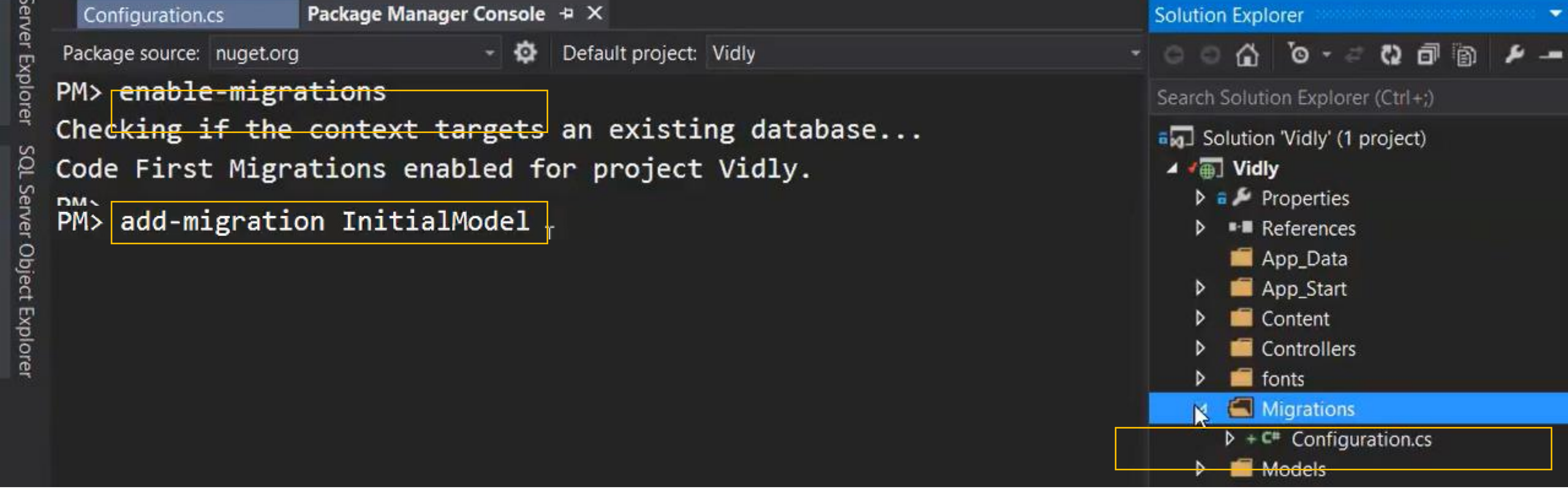
- Increased productivity
- Full versioning of database
- Much easier to build an integration test db

Code-first Workflow

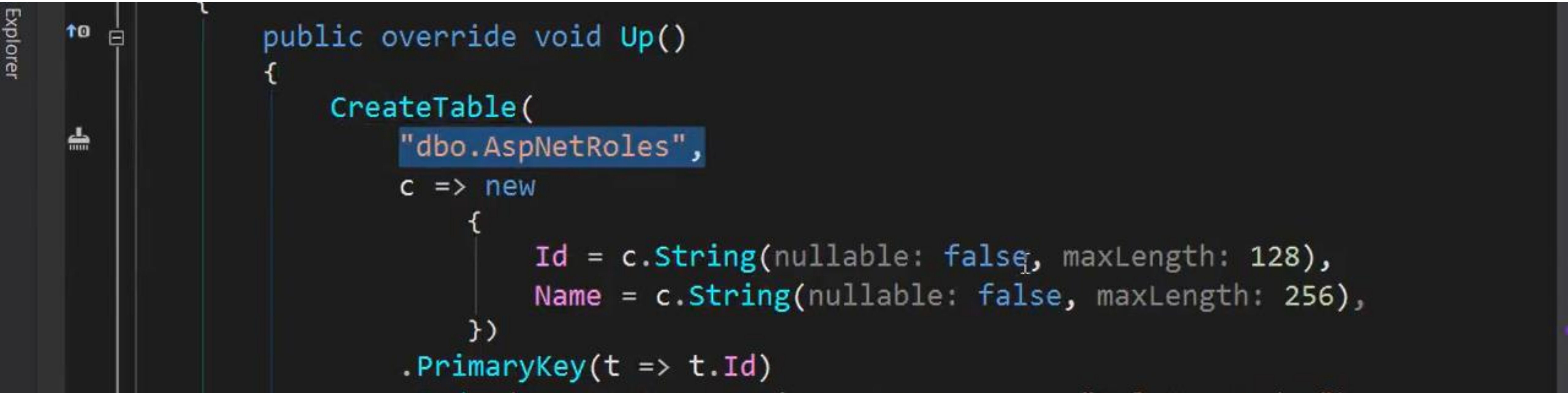
Code-First Migrations

Zouhair Rimale, Ph.D.

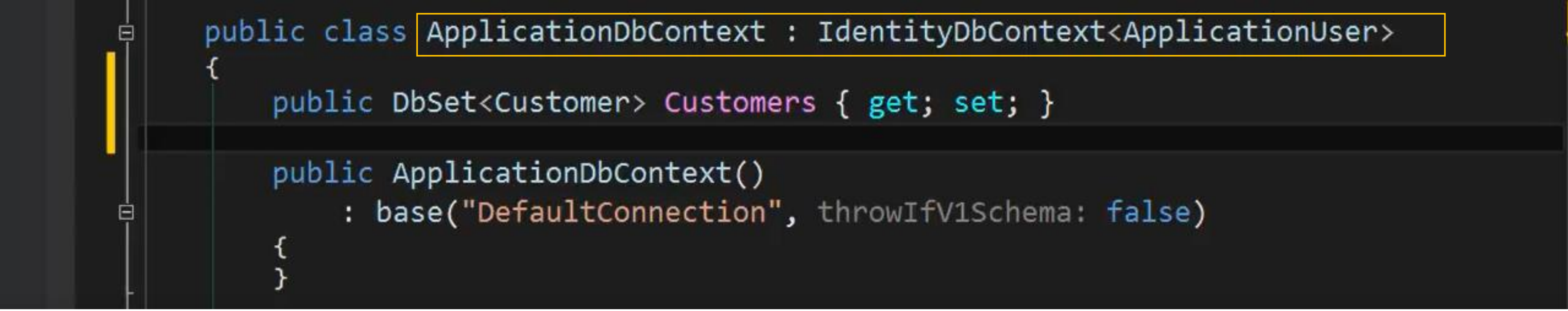
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



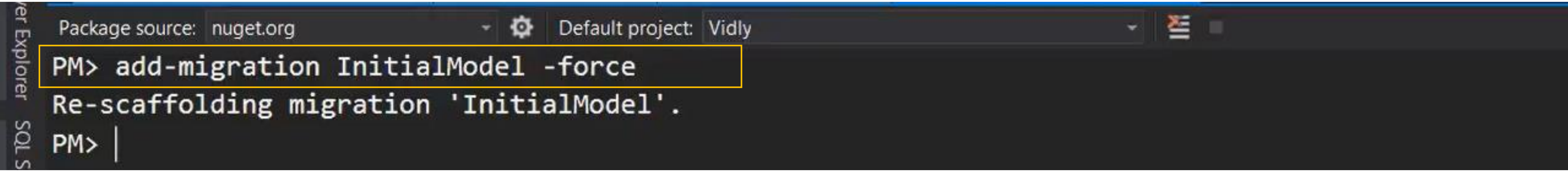
```
PM> enable-migrations
Checking if the context targets an existing database...
Code First Migrations enabled for project Vidly.
PM>
PM> add-migration InitialModel
```



```
        public override void Up()
        {
            CreateTable(
                "dbo.AspNetRoles",
                c => new
                    {
                        Id = c.String(nullable: false, maxLength: 128),
                        Name = c.String(nullable: false, maxLength: 256),
                    })
                .PrimaryKey(t => t.Id)
```



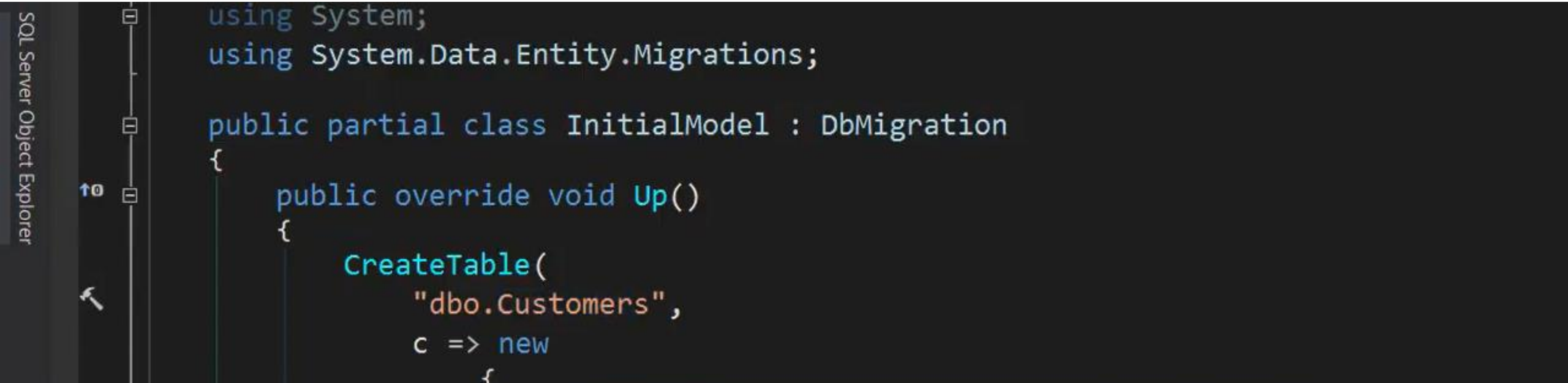
```
public class ApplicationDbContext : IdentityDbContext<ApplicationUser>
{
    public DbSet<Customer> Customers { get; set; }

    public ApplicationDbContext()
        : base("DefaultConnection", throwIfV1Schema: false)
    {
    }
```



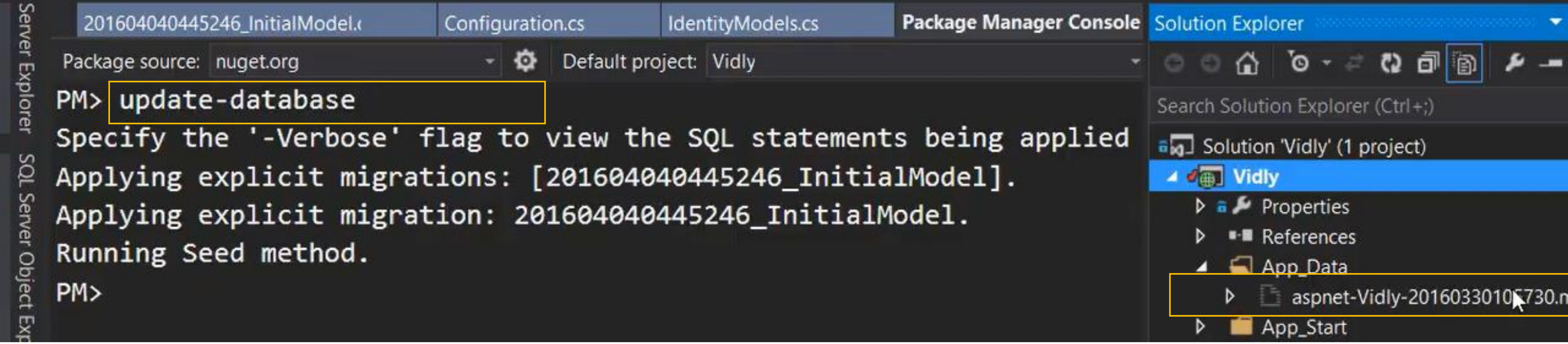
Package source: nuget.org

Default project: Vidly

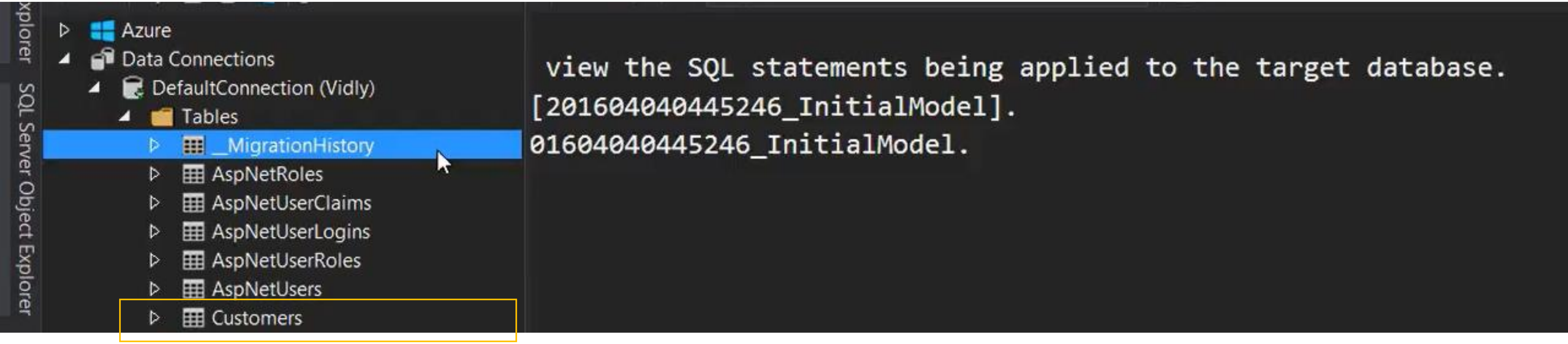
```
PM> add-migration InitialModel -force
Re-scaffolding migration 'InitialModel'.
PM>
```



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

public partial class InitialModel : DbMigration
{
    public override void Up()
    {
        CreateTable(
            "dbo.Customers",
            c => new
                {
```



```
PM> update-database
Specify the '-Verbose' flag to view the SQL statements being applied
Applying explicit migrations: [201604040445246_InitialModel].
Applying explicit migration: 201604040445246_InitialModel.
Running Seed method.
PM>
```



```
 view the SQL statements being applied to the target database.
[201604040445246_InitialModel].
01604040445246_InitialModel.
```

Changing the Model

Zouhair Rimale, Ph.D.

Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular

Domain Model

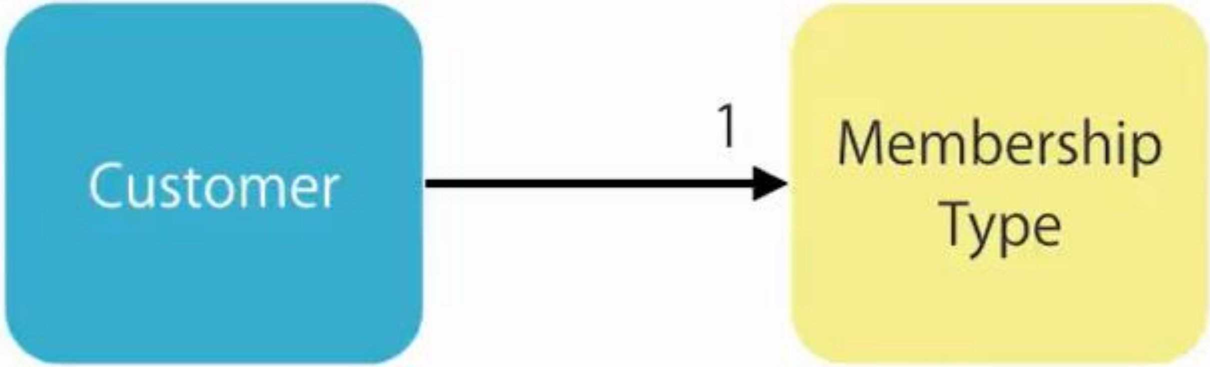
Customer

```
IsSubscribedToNewsletter
MembershipType
```

Rules

Membership	Sign Up Fee	Discount
Pay as You Go	Free	0%
Monthly	$30	10%
Quarterly	$90	15%
Annual	$300	20%

Domain Model



Customer

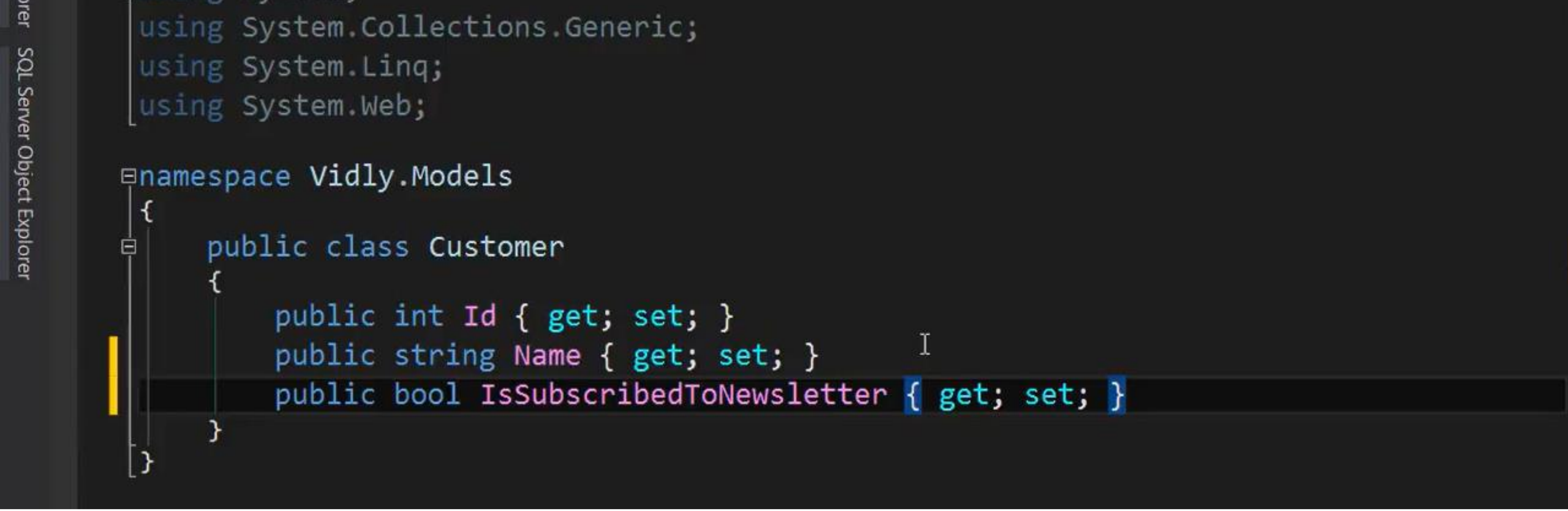
1

Membership Type

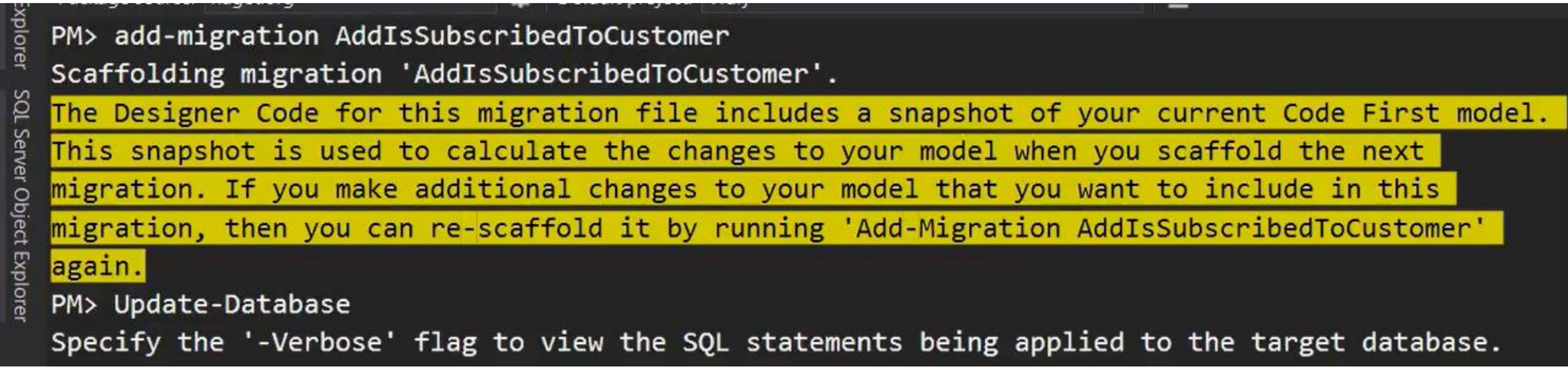
```
SignUpFee
Duration
DiscountRate
```

Aim for small migrations

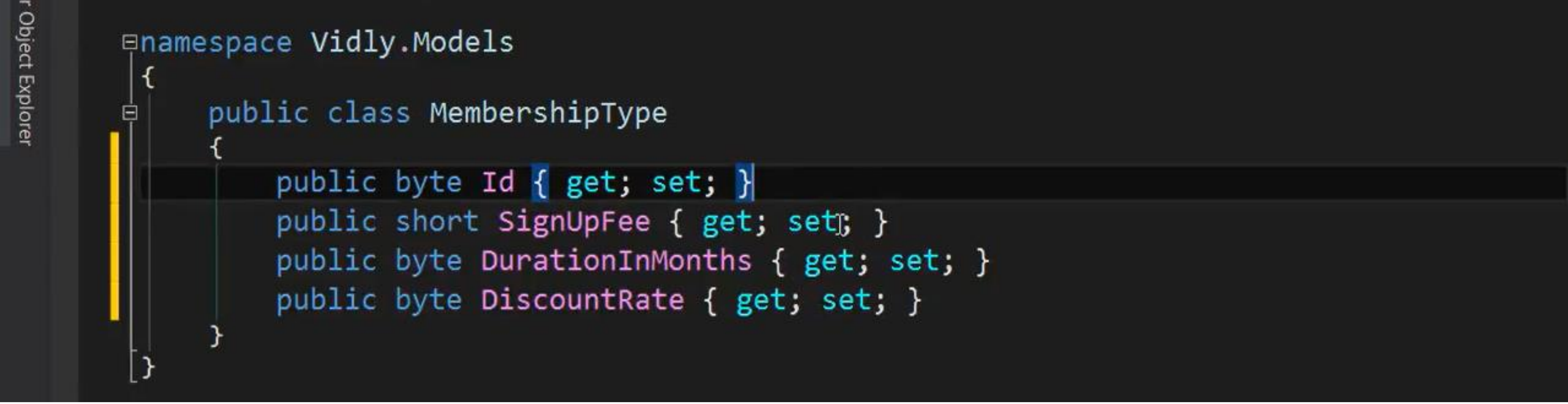
GIT COMMIT



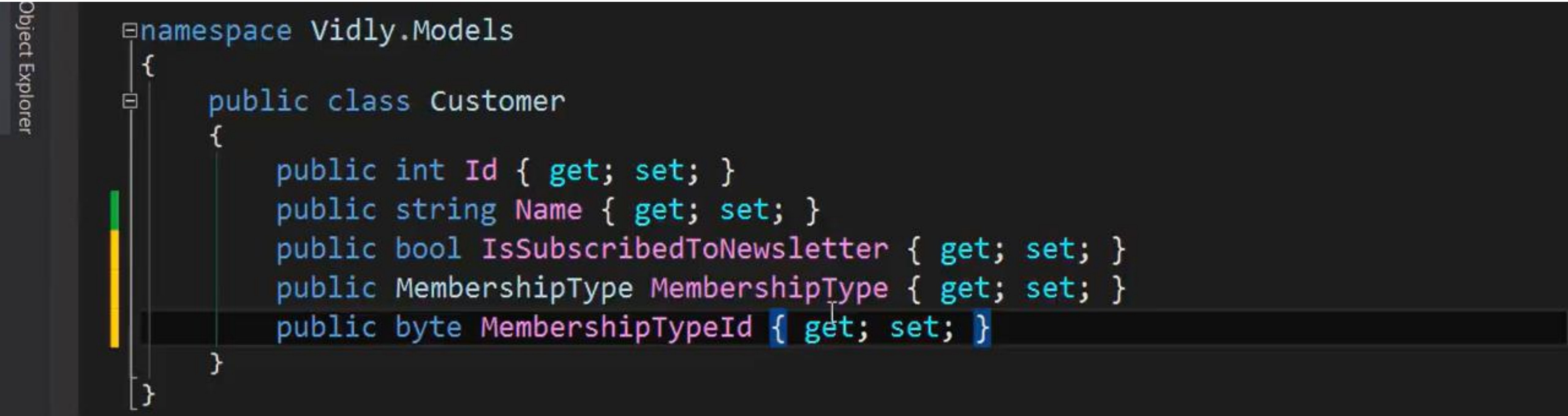
```
using System.Collections.Generic;
using System.Linq;
using System.Web;

namespace Vidly.Models
{
    public class Customer
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public bool IsSubscribedToNewsletter { get; set; }
    }
}
```



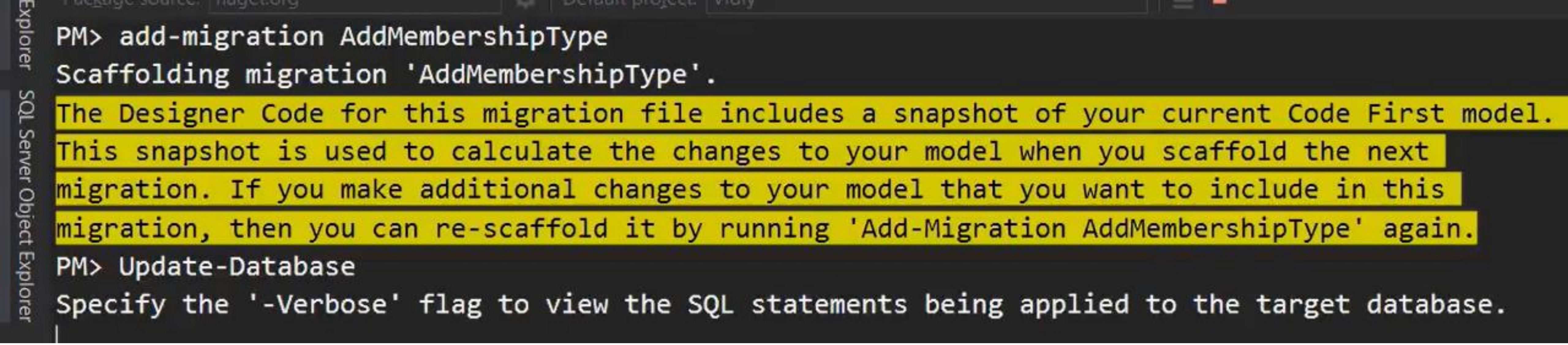
```
PM> add-migration AddIsSubscribedToCustomer
Scaffolding migration 'AddIsSubscribedToCustomer'.
The Designer Code for this migration file includes a snapshot of your current Code First model.
This snapshot is used to calculate the changes to your model when you scaffold the next
migration. If you make additional changes to your model that you want to include in this
migration, then you can re-scaffold it by running 'Add-Migration AddIsSubscribedToCustomer'
again.
PM> Update-Database
Specify the '-Verbose' flag to view the SQL statements being applied to the target database.
```



```
namespace Vidly.Models
{
    public class MembershipType
    {
        public byte Id { get; set; }
        public short SignUpFee { get; set; }
        public byte DurationInMonths { get; set; }
        public byte DiscountRate { get; set; }
    }
}
```



```
namespace Vidly.Models
{
    public class Customer
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public bool IsSubscribedToNewsletter { get; set; }
        public MembershipType MembershipType { get; set; }
        public byte MembershipTypeId { get; set; }
    }
}
```

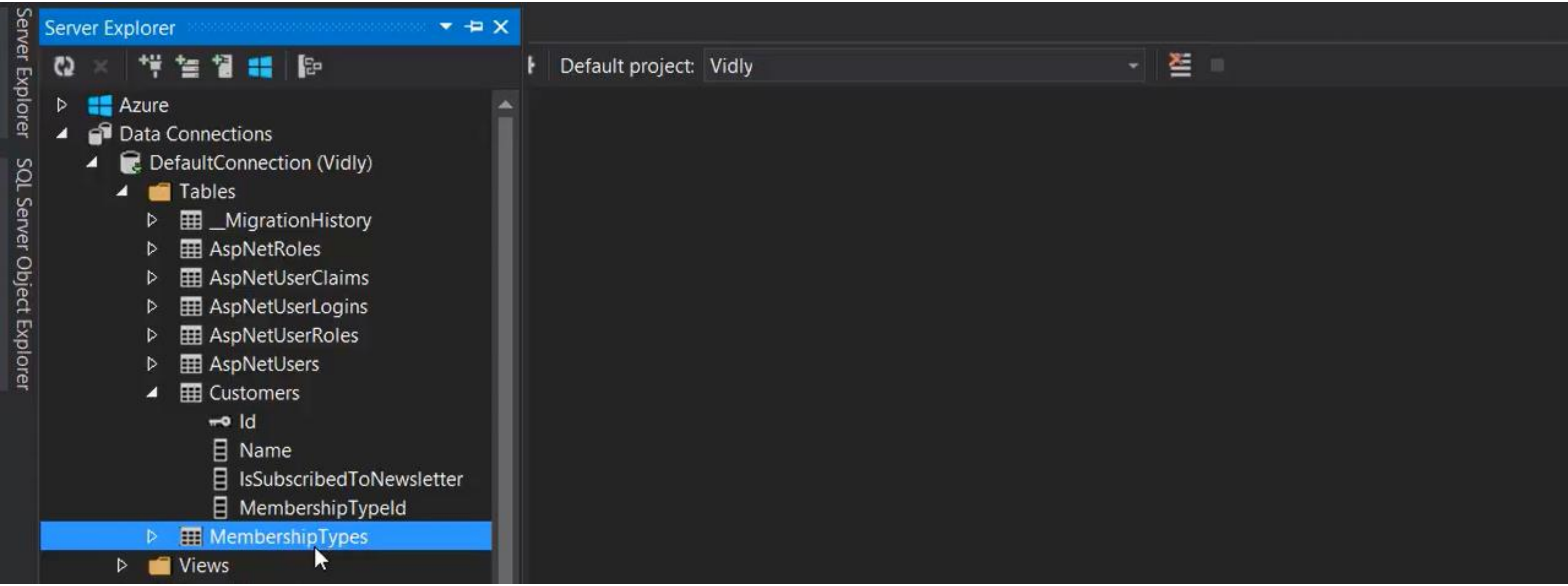


```
PM> add-migration AddMembershipType
Scaffolding migration 'AddMembershipType'.
The Designer Code for this migration file includes a snapshot of your current Code First model.
This snapshot is used to calculate the changes to your model when you scaffold the next
migration. If you make additional changes to your model that you want to include in this
migration, then you can re-scaffold it by running 'Add-Migration AddMembershipType' again.
PM> Update-Database
Specify the '-Verbose' flag to view the SQL statements being applied to the target database.
```

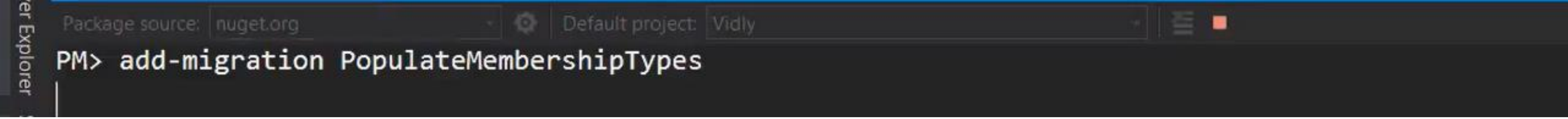
Seeding the Database

Zouhair Rimale, Ph.D.

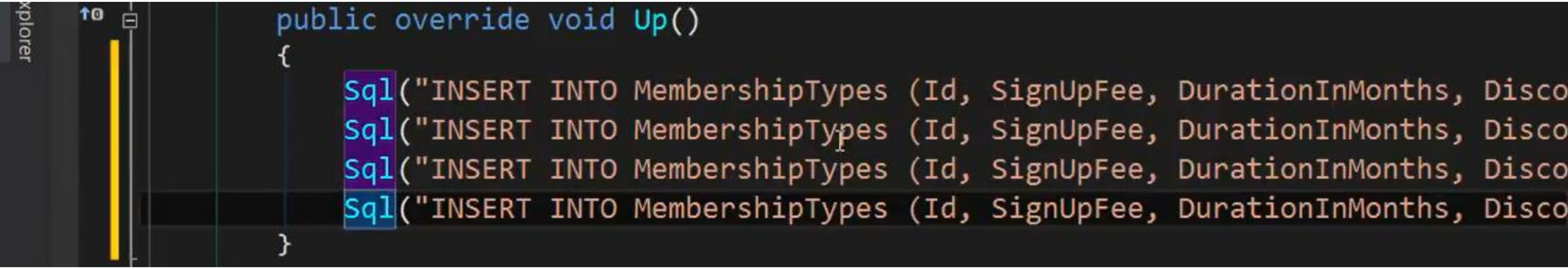
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



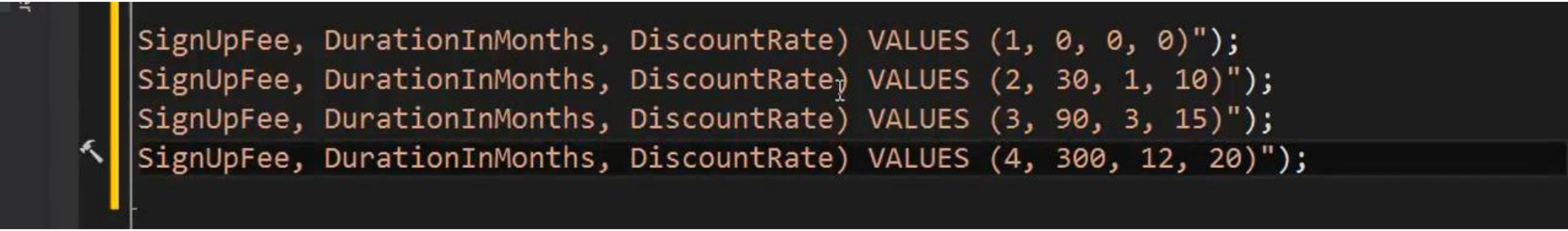
Default project: Vidly



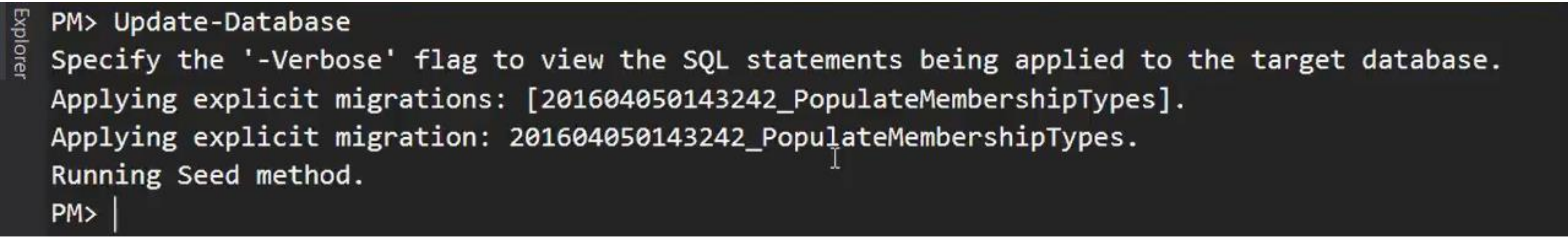
```
PM> add-migration PopulateMembershipTypes
```



```
        public override void Up()
        {
            Sql("INSERT INTO MembershipTypes (Id, SignUpFee, DurationInMonths, Disco
            Sql("INSERT INTO MembershipTypes (Id, SignUpFee, DurationInMonths, Disco
            Sql("INSERT INTO MembershipTypes (Id, SignUpFee, DurationInMonths, Disco
            Sql("INSERT INTO MembershipTypes (Id, SignUpFee, DurationInMonths, Disco
        }
```



```
SignUpFee, DurationInMonths, DiscountRate) VALUES (1, 0, 0, 0)");
SignUpFee, DurationInMonths, DiscountRate) VALUES (2, 30, 1, 10)");
SignUpFee, DurationInMonths, DiscountRate) VALUES (3, 90, 3, 15)");
SignUpFee, DurationInMonths, DiscountRate) VALUES (4, 300, 12, 20)");
```

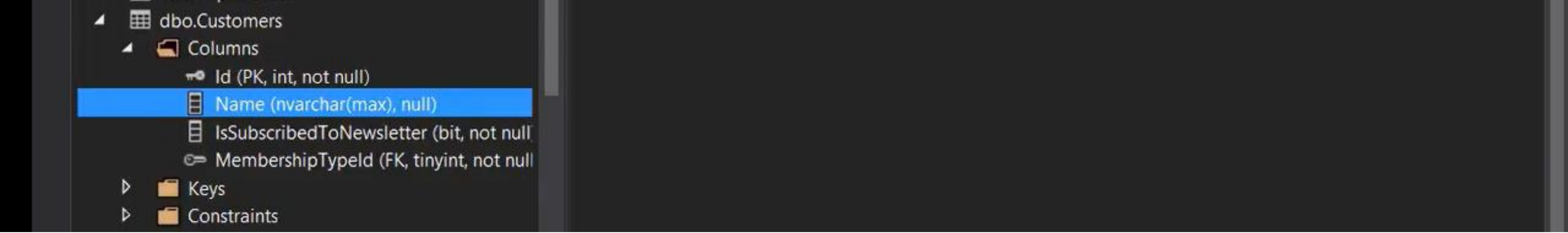


```
PM> Update-Database
Specify the '-Verbose' flag to view the SQL statements being applied to the target database.
Applying explicit migrations: [201604050143242_PopulateMembershipTypes].
Applying explicit migration: 201604050143242_PopulateMembershipTypes.
Running Seed method.
PM>
```

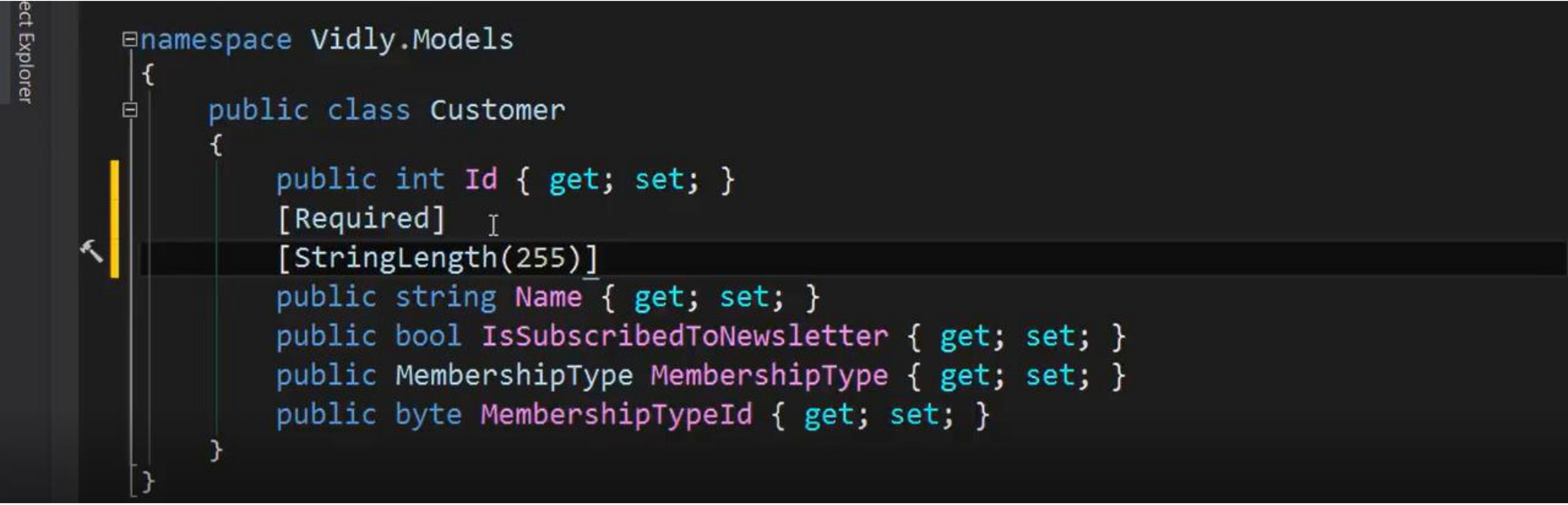
Overriding Conventions

Zouhair Rimale, Ph.D.

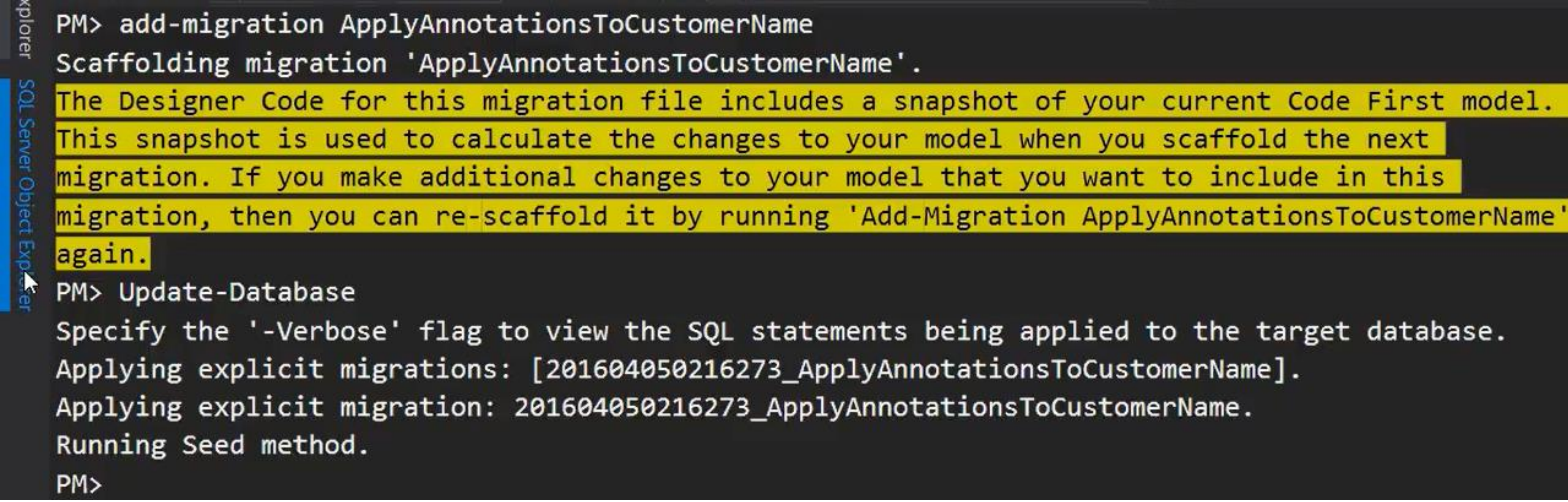
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



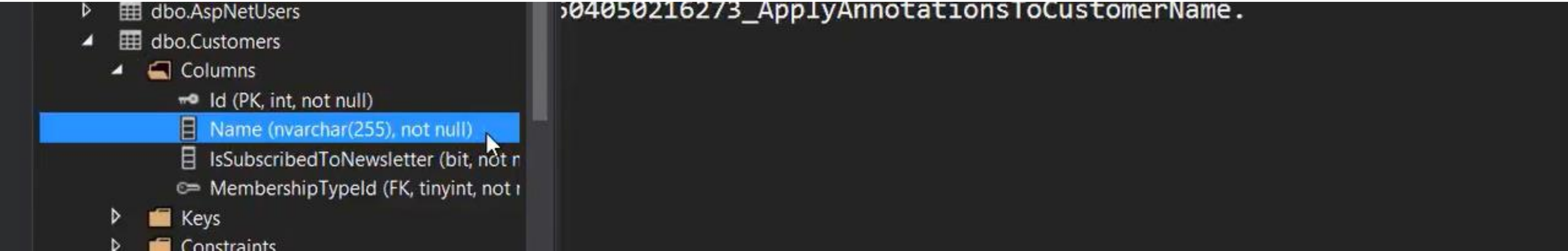
```
◢ dbo.Customers
    ◢ Columns
        Id (PK, int, not null)
        Name (nvarchar(max), null)
        IsSubscribedToNewsletter (bit, not null
        MembershipTypeId (FK, tinyint, not null
    ▷ Keys
    ▷ Constraints
```



```
namespace Vidly.Models
{
    public class Customer
    {
        public int Id { get; set; }
        [Required]
        [StringLength(255)]
        public string Name { get; set; }
        public bool IsSubscribedToNewsletter { get; set; }
        public MembershipType MembershipType { get; set; }
        public byte MembershipTypeId { get; set; }
    }
}
```



```
PM> add-migration ApplyAnnotationsToCustomerName
Scaffolding migration 'ApplyAnnotationsToCustomerName'.
The Designer Code for this migration file includes a snapshot of your current Code First model.
This snapshot is used to calculate the changes to your model when you scaffold the next
migration. If you make additional changes to your model that you want to include in this
migration, then you can re-scaffold it by running 'Add-Migration ApplyAnnotationsToCustomerName'
again.
PM> Update-Database
Specify the '-Verbose' flag to view the SQL statements being applied to the target database.
Applying explicit migrations: [201604050216273_ApplyAnnotationsToCustomerName].
Applying explicit migration: 201604050216273_ApplyAnnotationsToCustomerName.
Running Seed method.
PM>
```

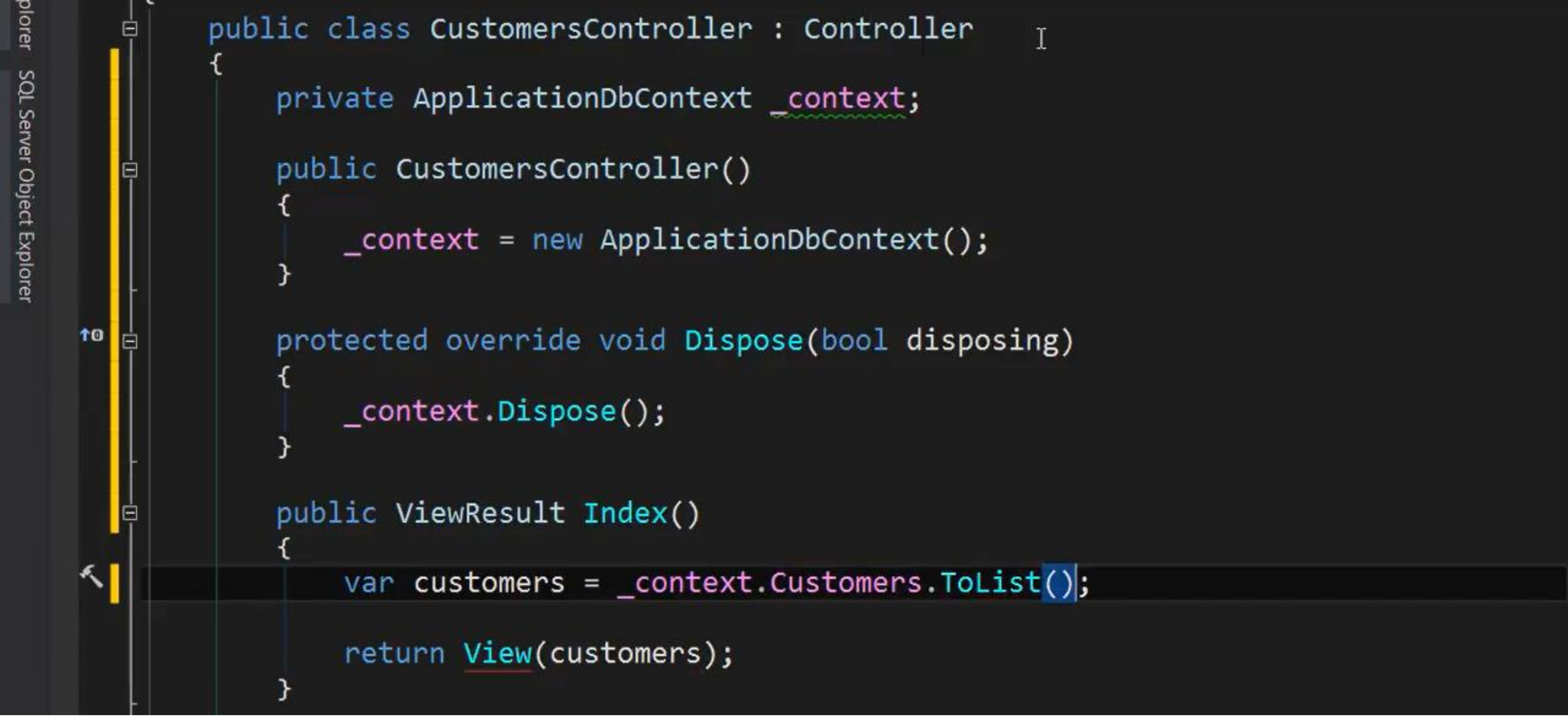

dbo.AspNetUsers
dbo.Customers
Columns
Id (PK, int, not null)
Name (nvarchar(255), not null)
IsSubscribedToNewsletter (bit, not n
MembershipTypeId (FK, tinyint, not r
Keys
Constraints

```
50405021627 3_ApplyAnnotationsToCustomerName.
```

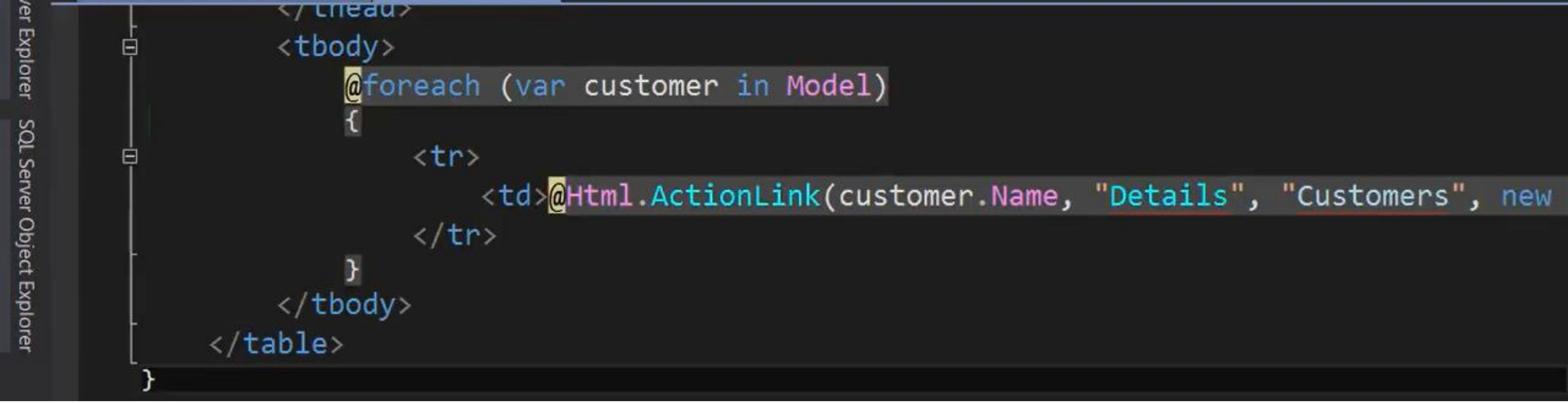
Querying Objects

Zouhair Rimale, Ph.D.

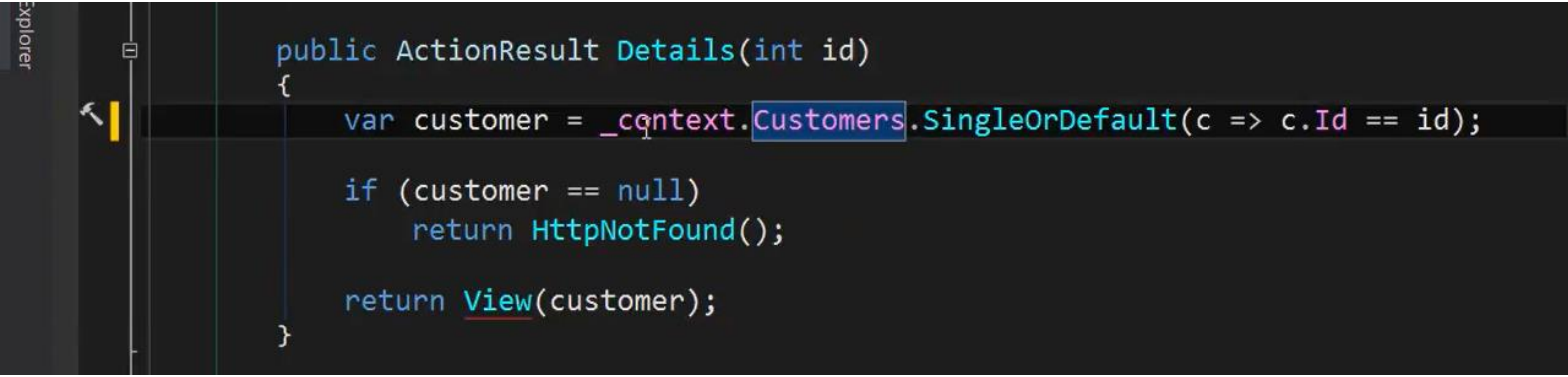
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



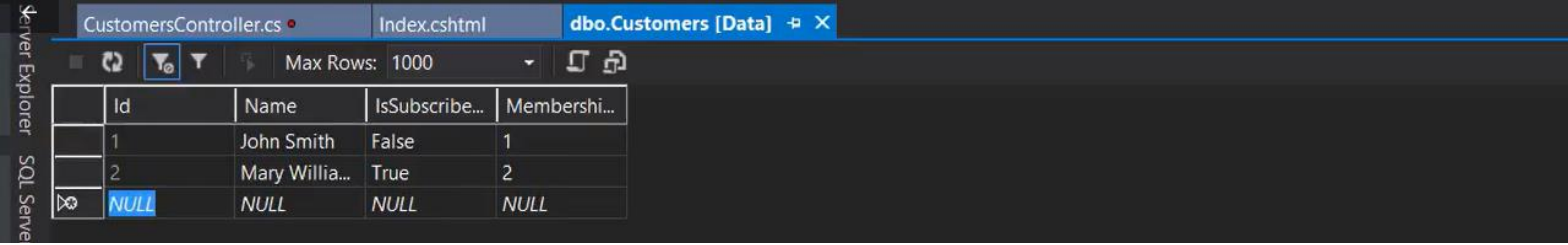
```
    public class CustomersController : Controller
    {
        private ApplicationDbContext _context;

        public CustomersController()
        {
            _context = new ApplicationDbContext();
        }

        protected override void Dispose(bool disposing)
        {
            _context.Dispose();
        }

        public ViewResult Index()
        {
            var customers = _context.Customers.ToList();

            return View(customers);
        }
```



```
        </thead>
        <tbody>
            @foreach (var customer in Model)
            {
                <tr>
                    <td>@Html.ActionLink(customer.Name, "Details", "Customers", new
                </tr>
            }
        </tbody>
    </table>
}
```

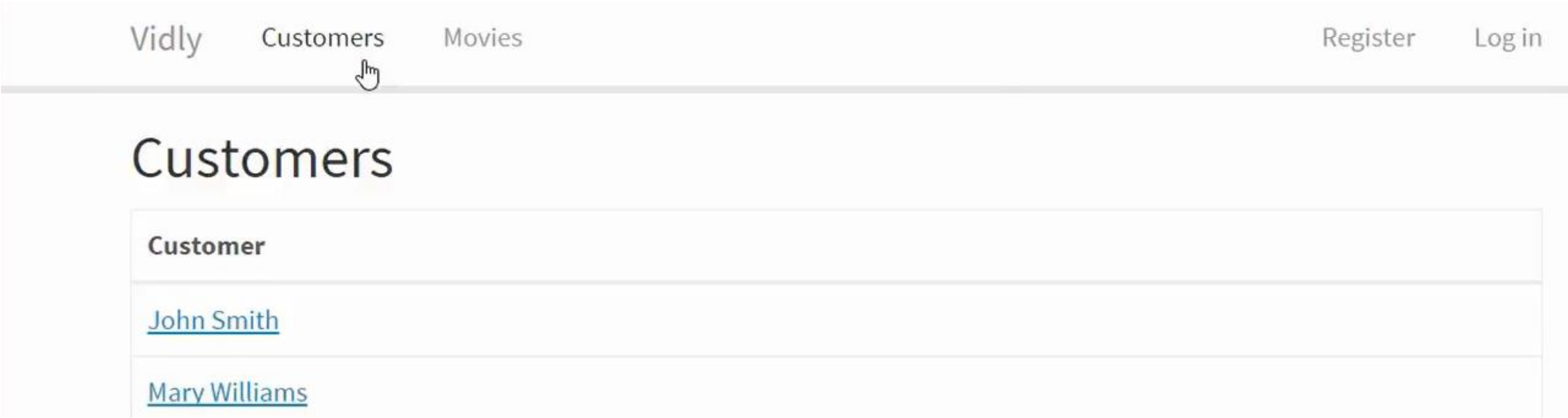


```
public ActionResult Details(int id)
{
    var customer = _context.Customers.SingleOrDefault(c => c.Id == id);

    if (customer == null)
        return HttpNotFound();

    return View(customer);
}
```



Max Rows: 1000

Id	Name	IsSubscribe...	Membershi...
1	John Smith	False	1
2	Mary Willia...	True	2
NULL	NULL	NULL	NULL



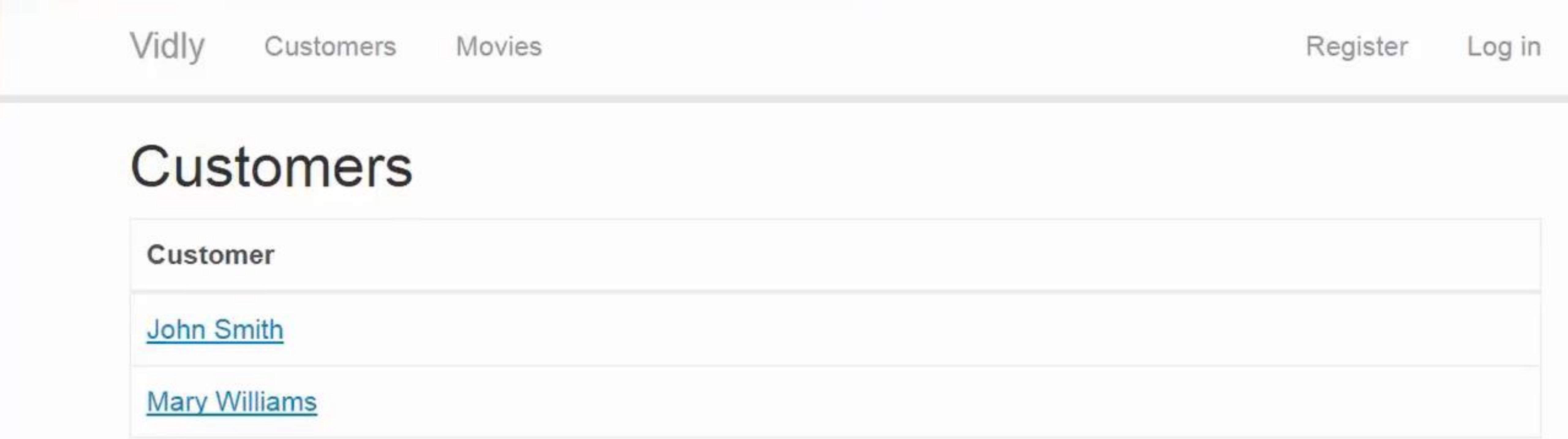
Customers

Customer
John Smith
Mary Williams

Eager Loading

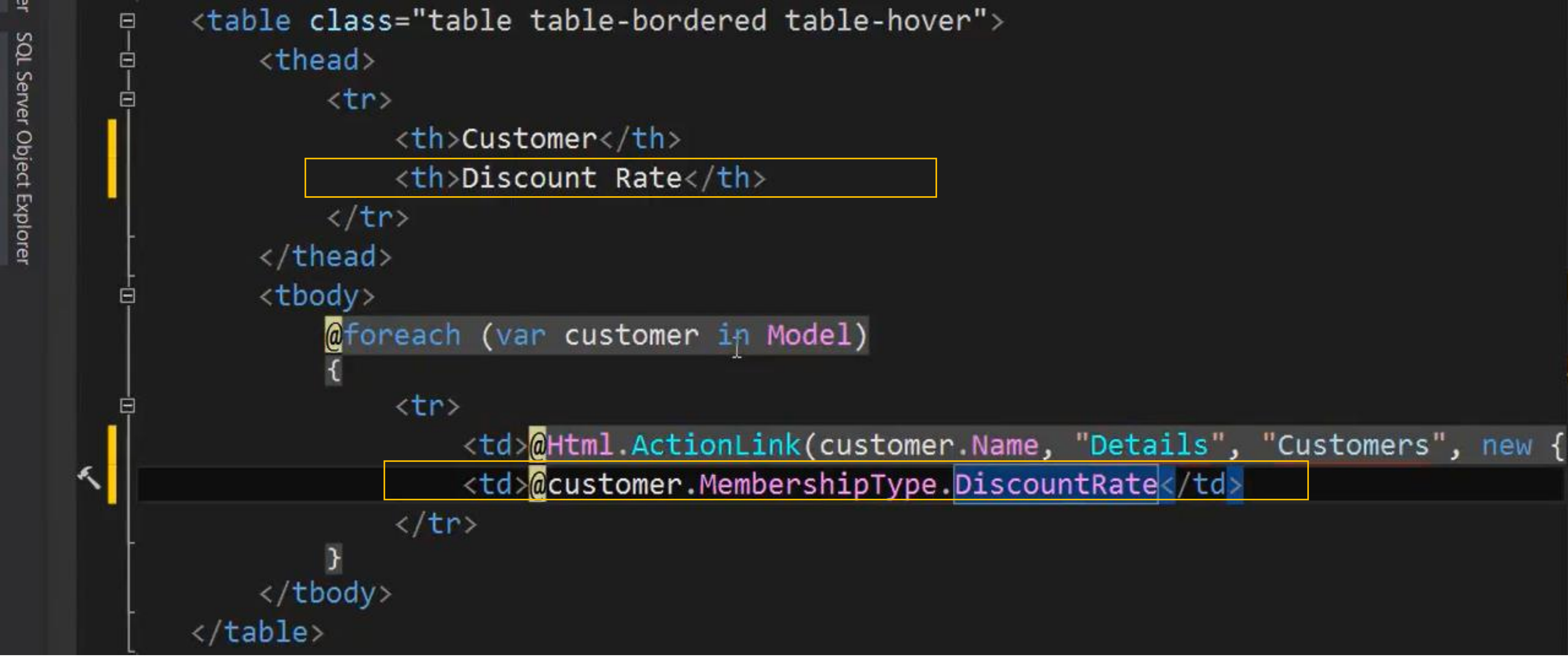
Zouhair Rimale, Ph.D.

Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular

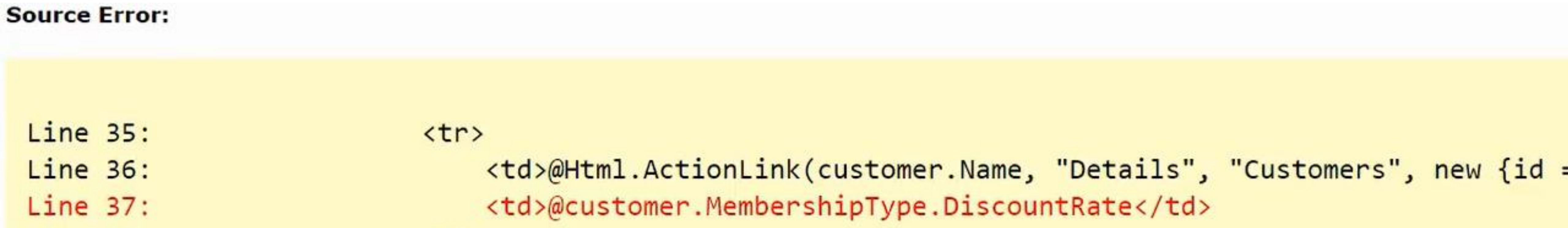


Customers

Customer
John Smith
Mary Williams

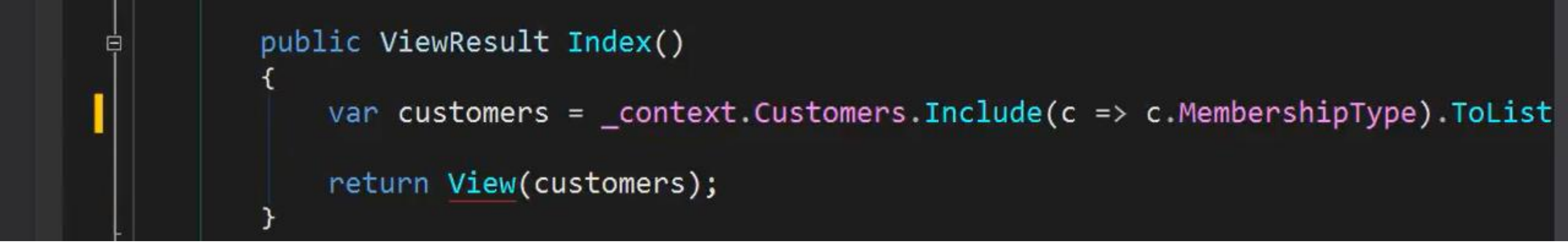


```
<table class="table table-bordered table-hover">
    <thead>
        <tr>
            <th>Customer</th>
            <th>Discount Rate</th>
        </tr>
    </thead>
    <tbody>
        @foreach (var customer in Model)
        {
            <tr>
                <td>@Html.ActionLink(customer.Name, "Details", "Customers", new {
                <td>@customer.MembershipType.DiscountRate</td>
            </tr>
        }
    </tbody>
</table>
```

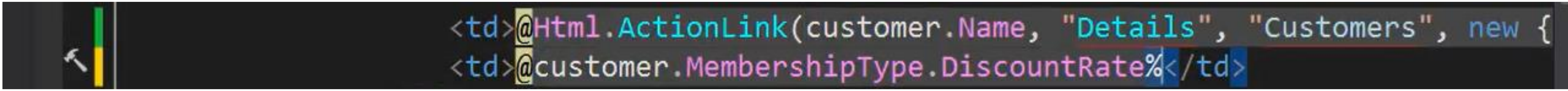


Source Error:

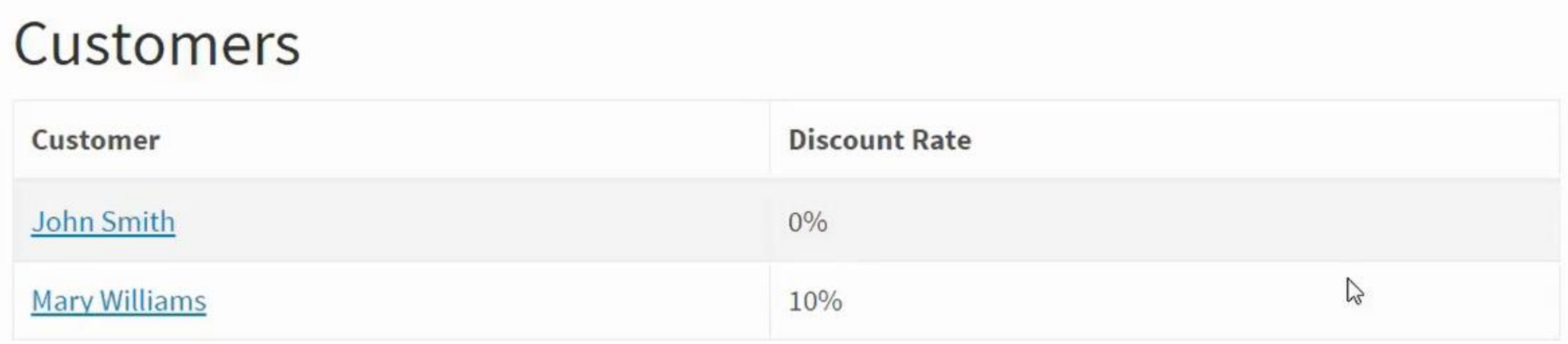
```
Line 35:                <tr>
Line 36:                    <td>@Html.ActionLink(customer.Name, "Details", "Customers", new {id =
Line 37:                    <td>@customer.MembershipType.DiscountRate</td>
```



```
public ViewResult Index()
{
    var customers = _context.Customers.Include(c => c.MembershipType).ToList

    return View(customers);
}
```



```
<td>@Html.ActionLink(customer.Name, "Details", "Customers", new {
<td>@customer.MembershipType.DiscountRate%</td>
```



Customers

Customer	Discount Rate
John Smith	0%
Mary Williams	10%

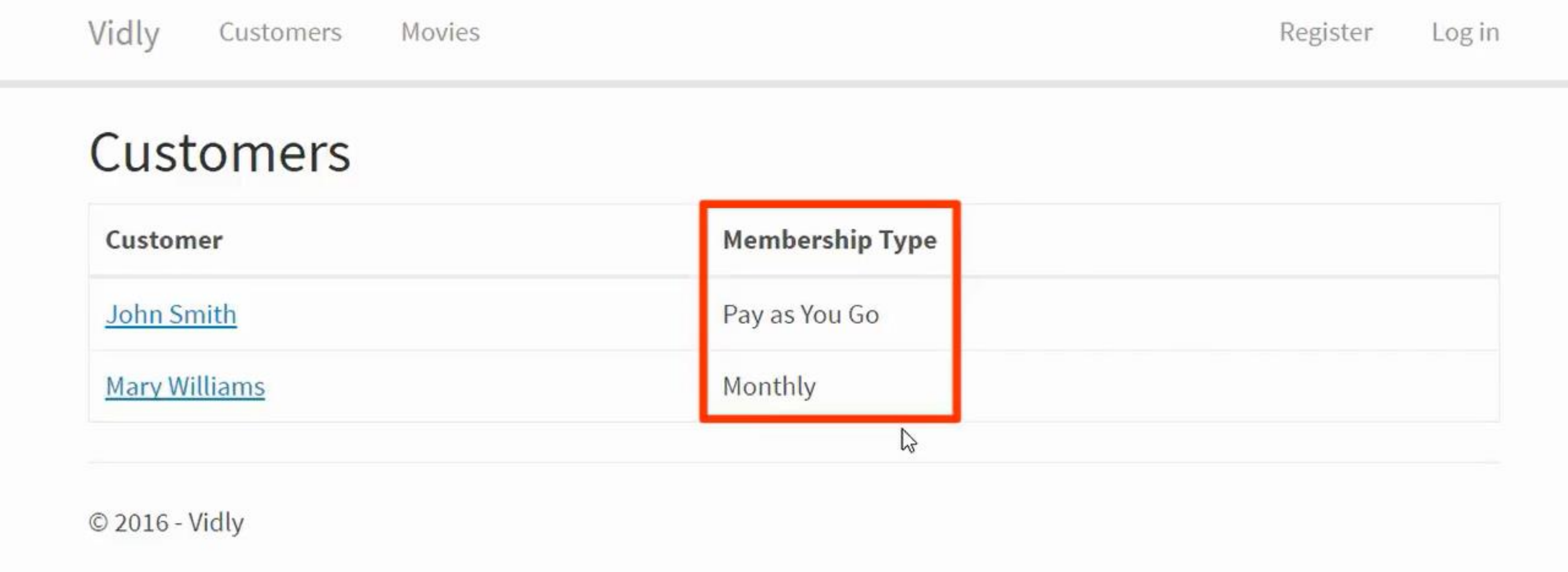
Package Manger Console

Zouhair Rimale, Ph.D.

Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular

Exercises

Exercise 1



Customers

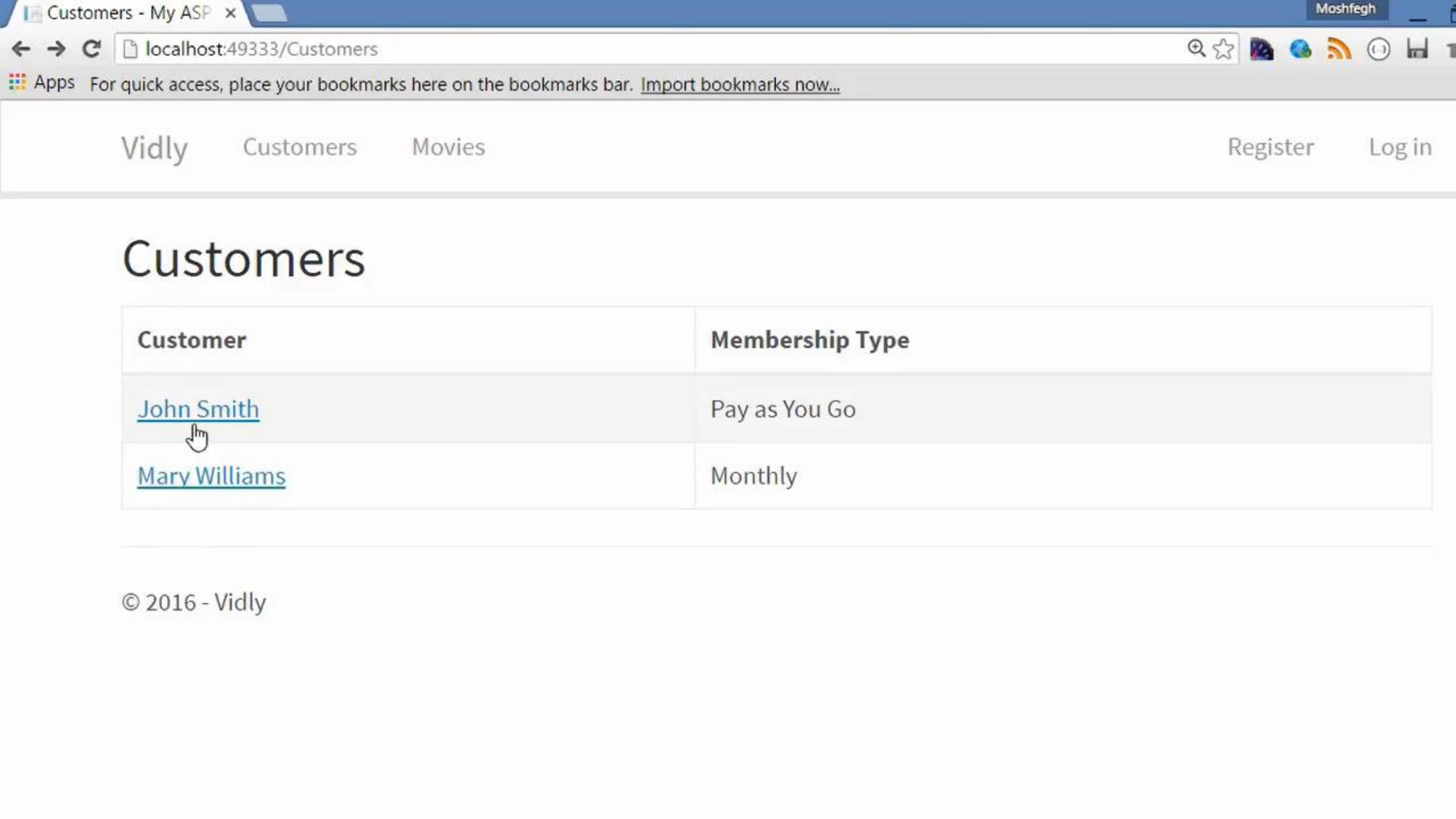
Customer	Membership Type
John Smith	Pay as You Go
Mary Williams	Monthly

© 2016 - Vidly

Hints

- Add **Name** to **MembershipType**
- Update the database
- Create another migration to update existing records

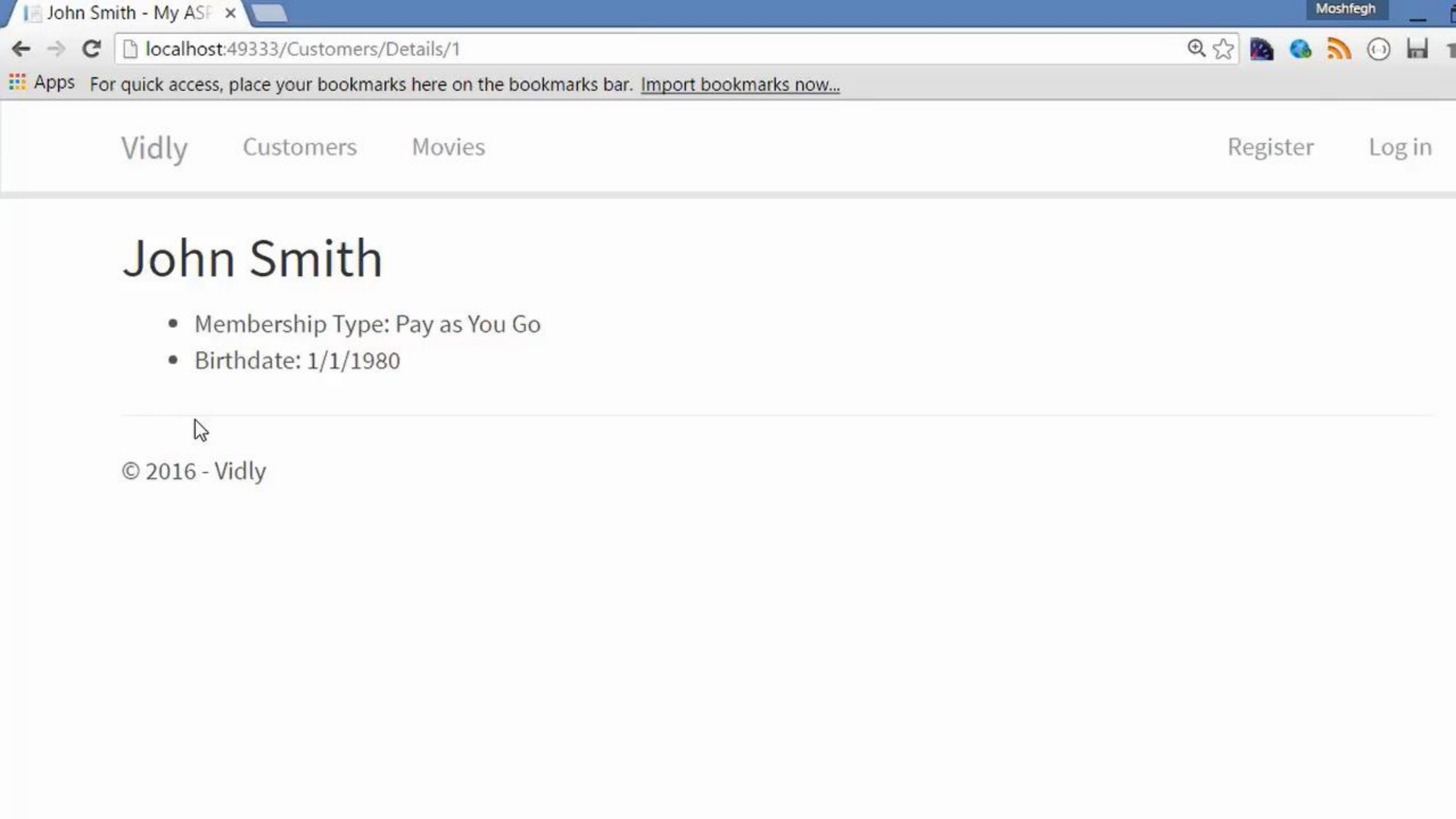
Exercise 2


Customers - My ASP
Moshfegh
localhost:49333/Customers
Apps
For quick access, place your bookmarks here on the bookmarks bar.
Import bookmarks now...
Vidly
Customers
Movies
Register
Log in

Customers

Customer	Membership Type
John Smith	Pay as You Go
Mary Williams	Monthly

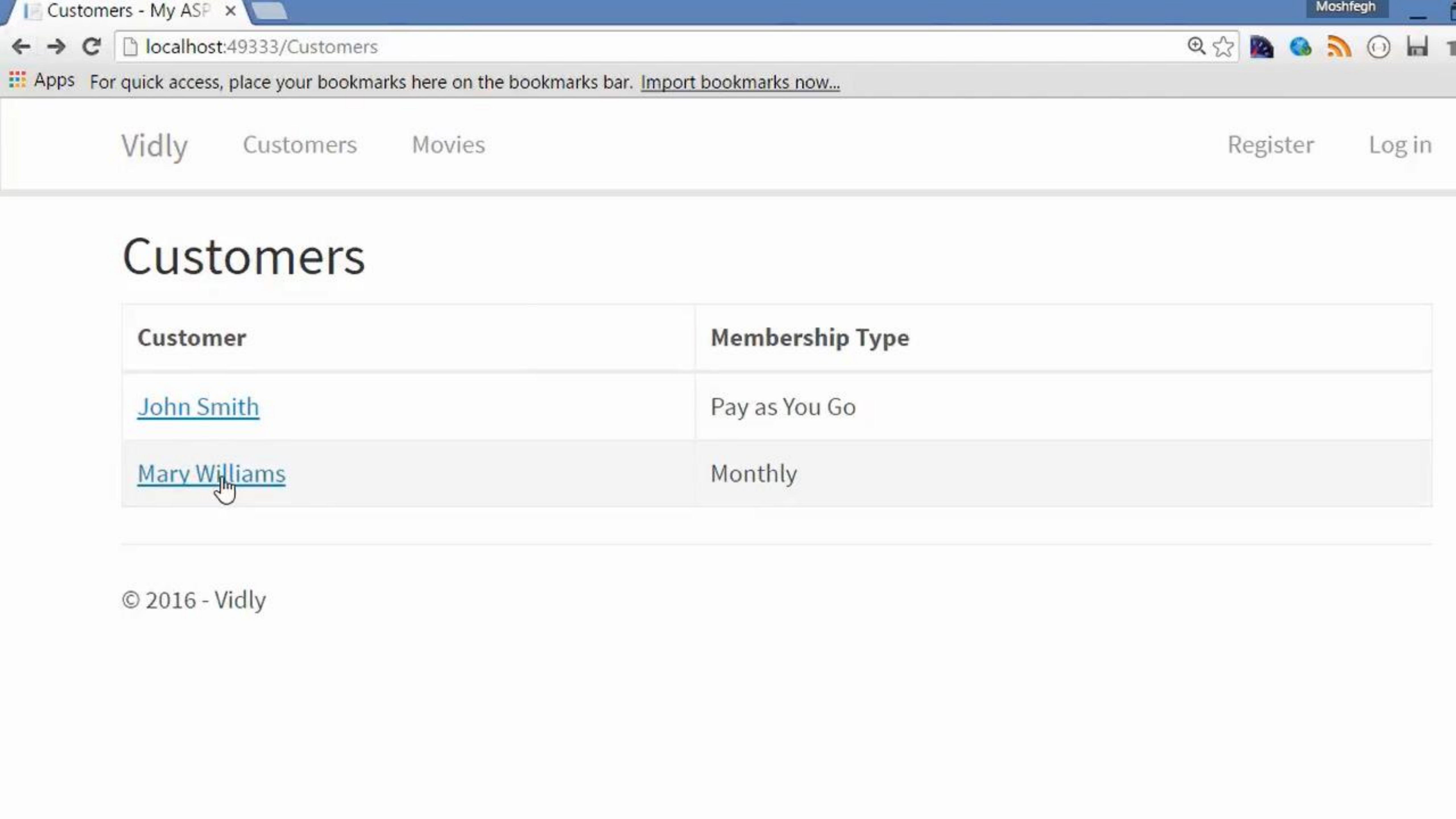
© 2016 - Vidly


John Smith - My ASP
Moshfegh
localhost:49333/Customers/Details/1
Apps
For quick access, place your bookmarks here on the bookmarks bar.
Import bookmarks now...
Vidly
Customers
Movies
Register
Log in

John Smith

- Membership Type: Pay as You Go
- Birthdate: 1/1/1980

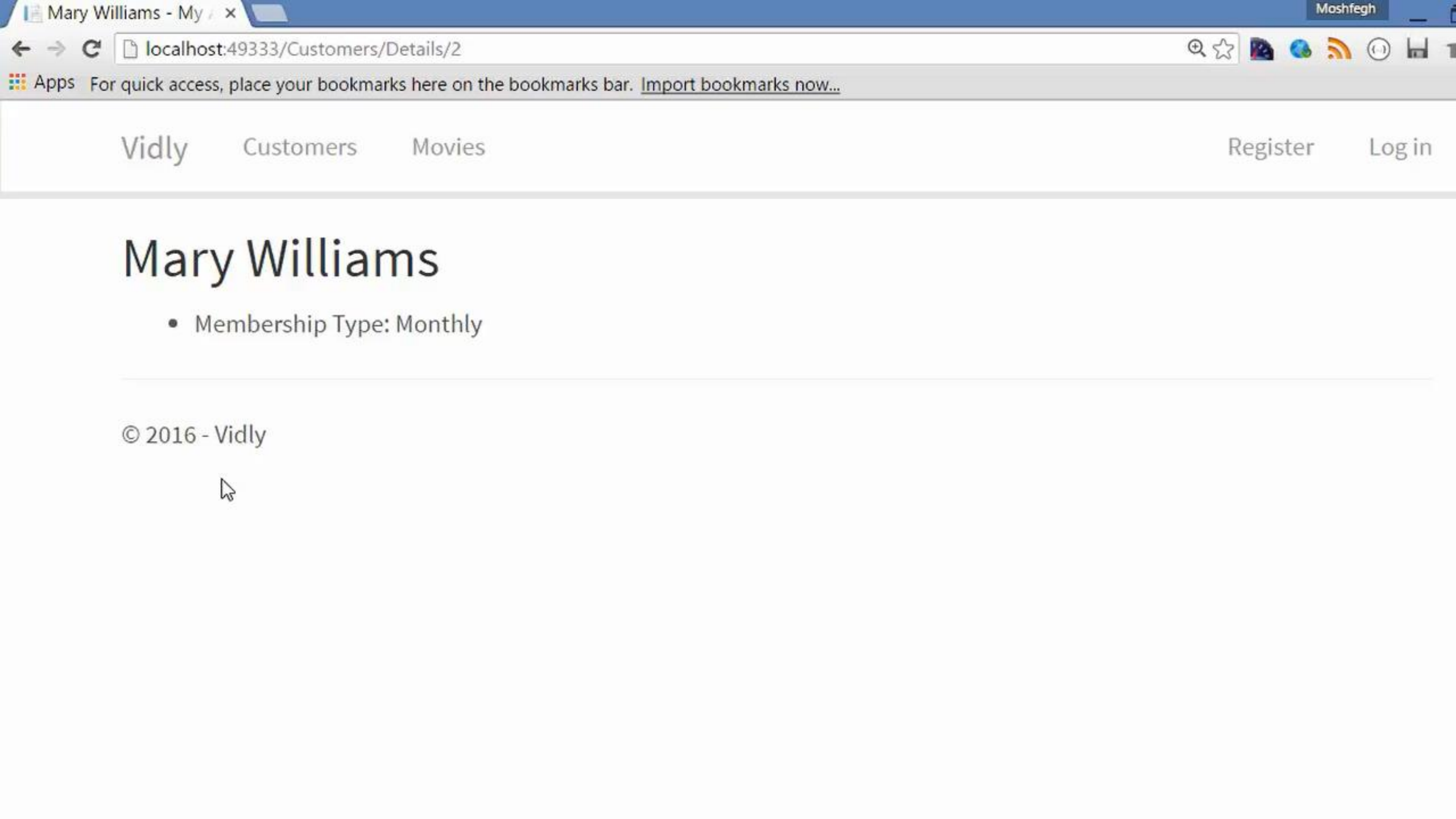
© 2016 - Vidly


Customers - My ASP
Moshfegh
localhost:49333/Customers
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Vidly
Customers
Movies
Register
Log in

Customers

Customer	Membership Type
John Smith	Pay as You Go
Mary Williams	Monthly

© 2016 - Vidly

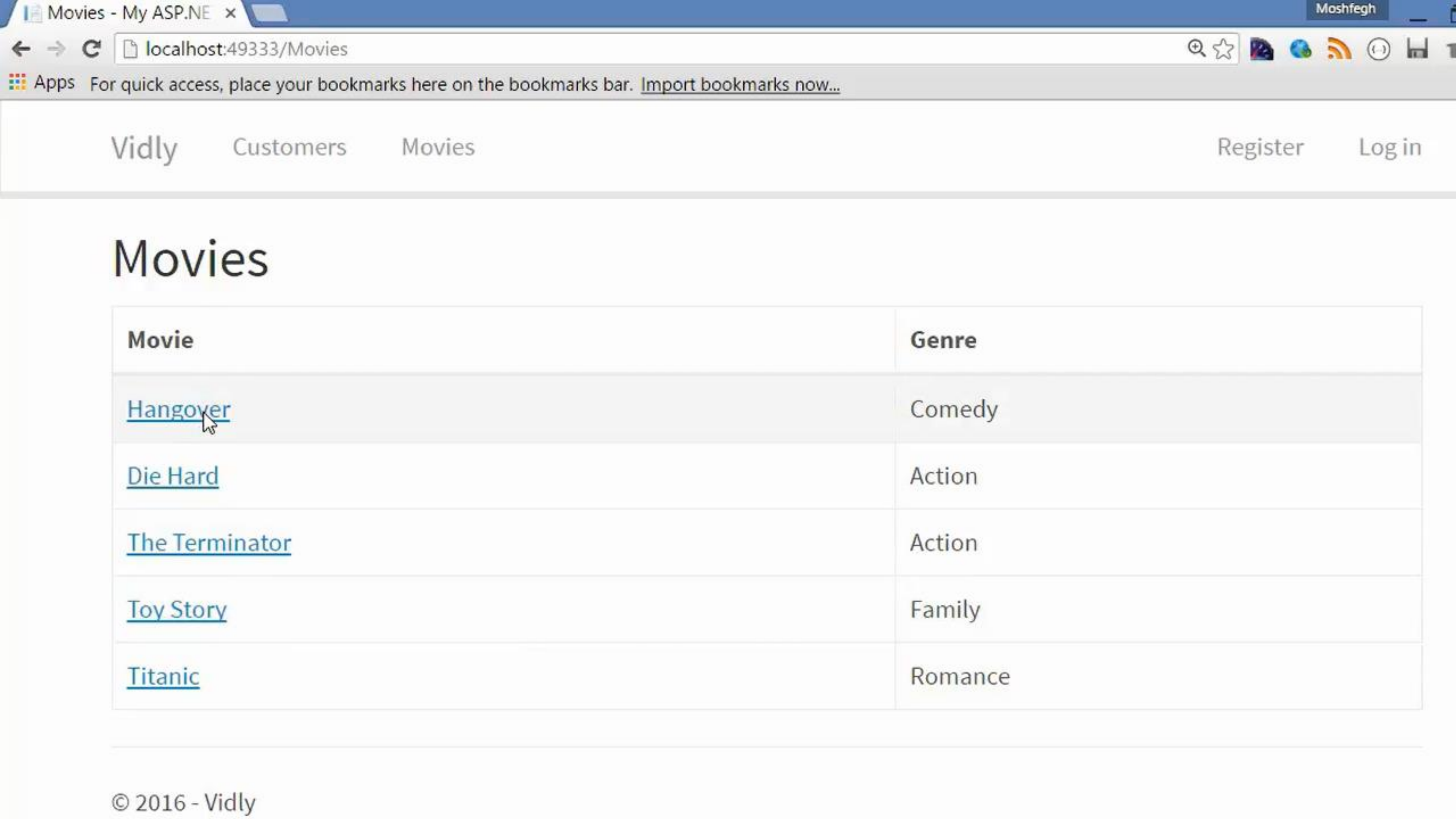

Mary Williams - My
Moshfegh
localhost:49333/Customers/Details/2
Apps
For quick access, place your bookmarks here on the bookmarks bar.
Import bookmarks now...
Vidly
Customers
Movies
Register
Log in

Mary Williams

- Membership Type: Monthly

© 2016 - Vidly

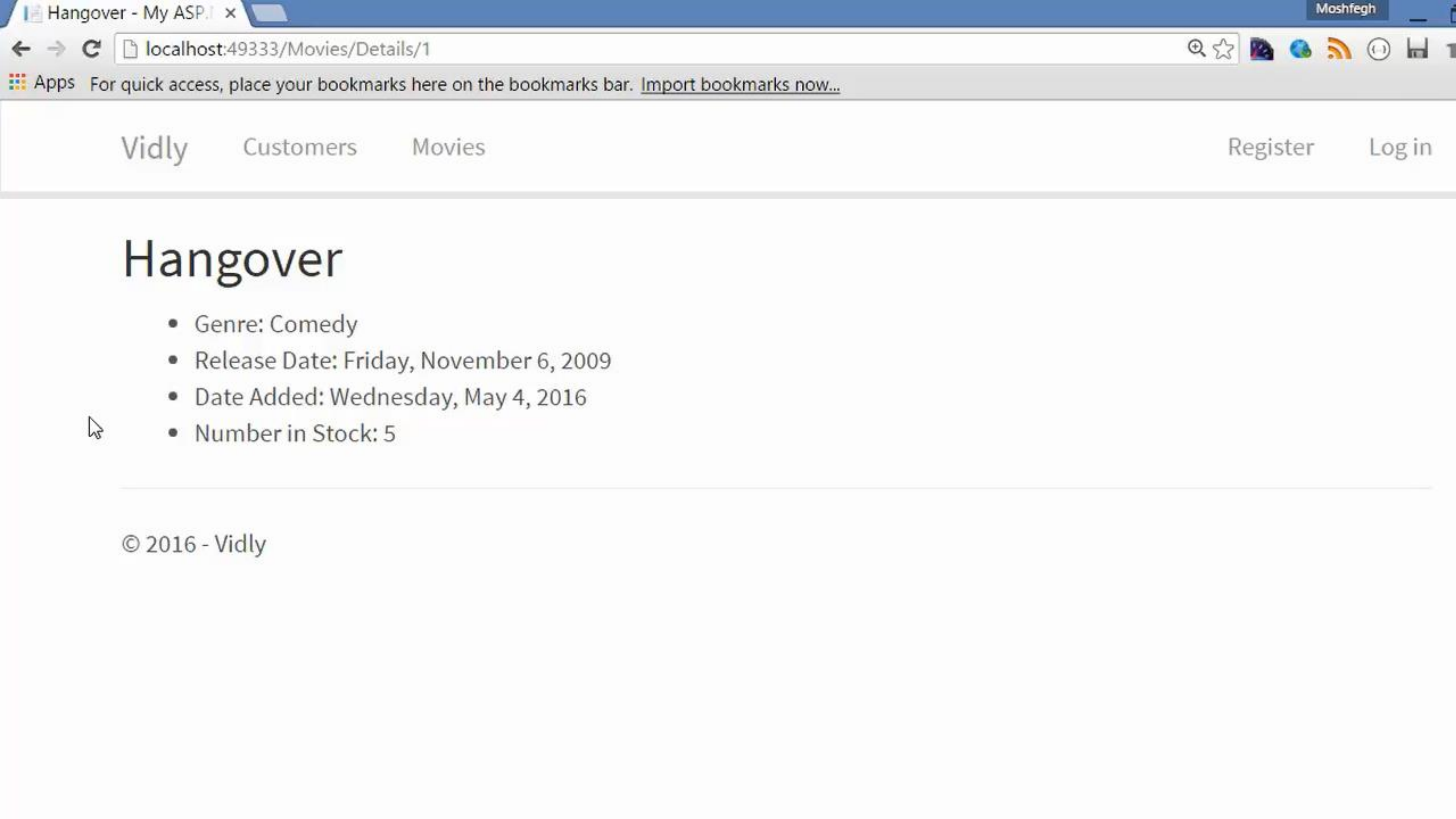
Exercise 3


Movies - My ASP.NE
Moshfegh
localhost:49333/Movies
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Vidly
Customers
Movies
Register
Log in

Movies

Movie	Genre
Hangover	Comedy
Die Hard	Action
The Terminator	Action
Toy Story	Family
Titanic	Romance

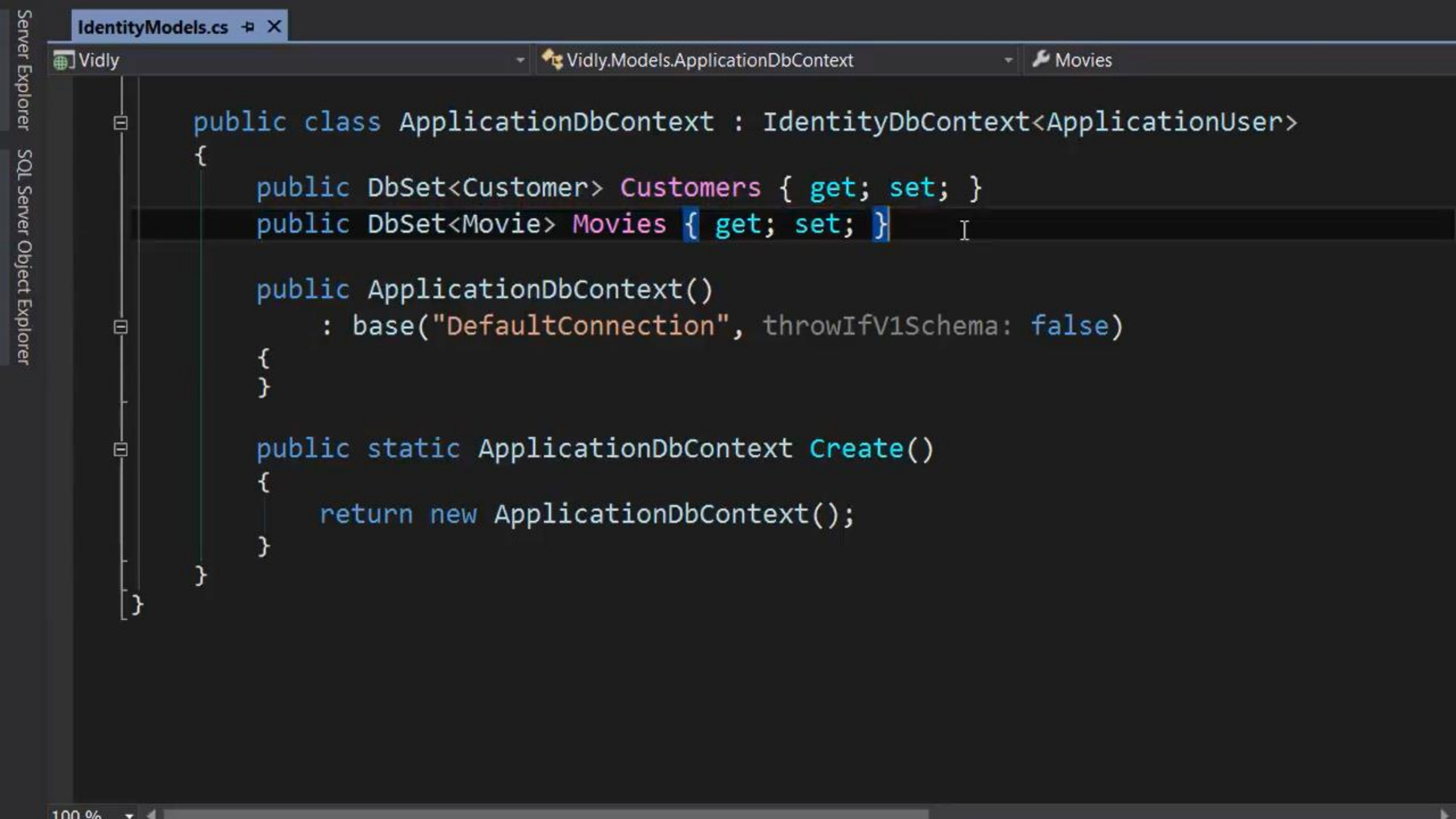
© 2016 - Vidly


Hangover - My ASP.
Moshfegh
localhost:49333/Movies/Details/1
Apps
For quick access, place your bookmarks here on the bookmarks bar.
Import bookmarks now...
Vidly
Customers
Movies
Register
Log in

Hangover

- Genre: Comedy
- Release Date: Friday, November 6, 2009
- Date Added: Wednesday, May 4, 2016
- Number in Stock: 5

© 2016 - Vidly



```
    public class ApplicationDbContext : IdentityDbContext<ApplicationUser>
    {
        public DbSet<Customer> Customers { get; set; }
        public DbSet<Movie> Movies { get; set; }

        public ApplicationDbContext()
            : base("DefaultConnection", throwIfV1Schema: false)
        {
        }

        public static ApplicationDbContext Create()
        {
            return new ApplicationDbContext();
        }
    }
}
```

Hints

- Add the additional properties in **Movie**
- Populate **Genres** table with a migration
- Modify **MoviesController** and its views

Summary

Working with Data

Summary

- Code-first Migrations
- Seeding the Database
- Overriding Conventions
- Querying Objects
- Eager Loading

Next

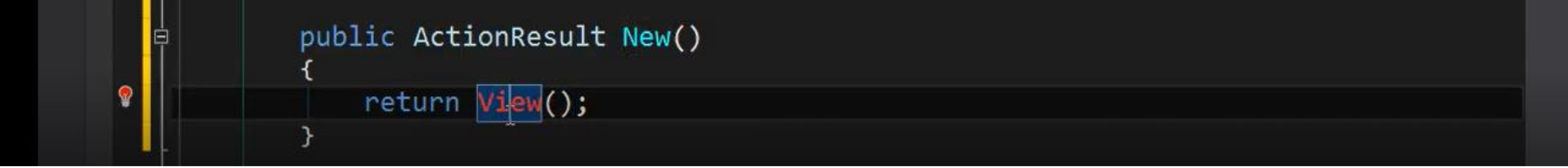
Building Forms

Building Forms

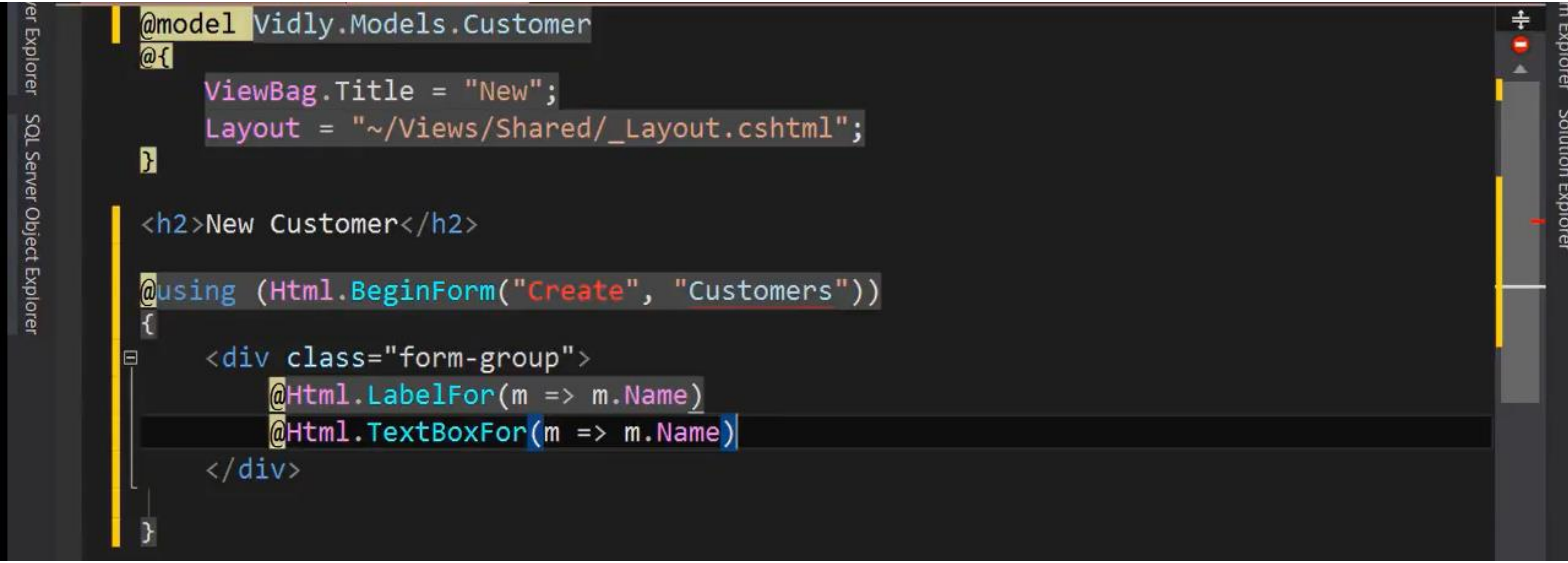
The Markup

Zouhair Rimale, Ph.D.

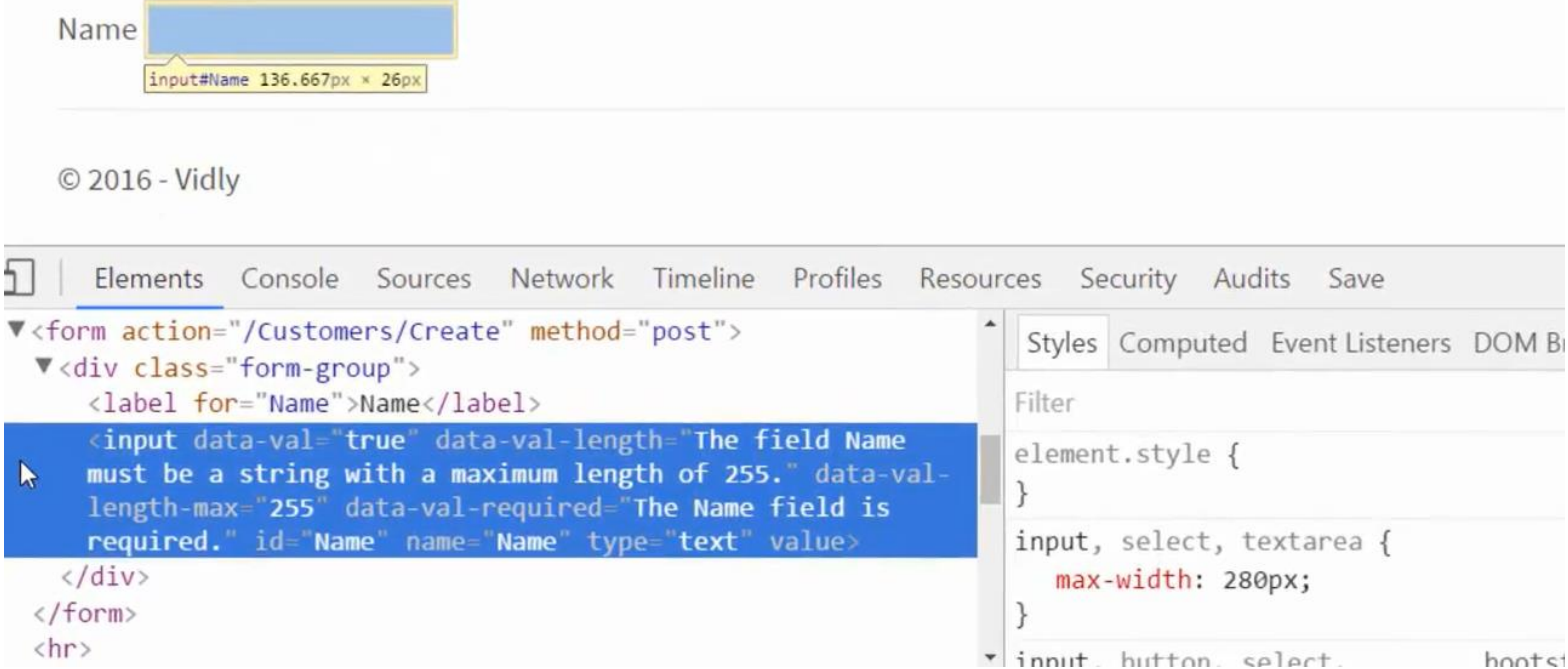
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



```
public ActionResult New()
{
    return View();
}
```

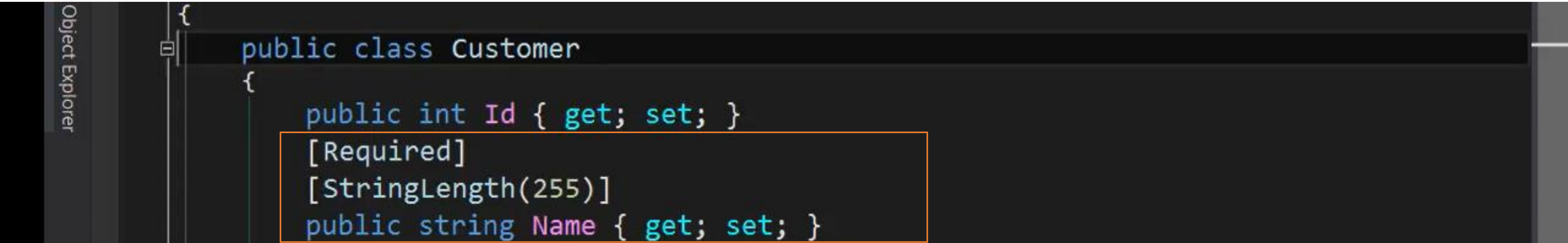


```
@model Vidly.Models.Customer
@{
    ViewBag.Title = "New";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<h2>New Customer</h2>

@using (Html.BeginForm("Create", "Customers"))
{
    <div class="form-group">
        @Html.LabelFor(m => m.Name)
        @Html.TextBoxFor(m => m.Name)
    </div>

}
```

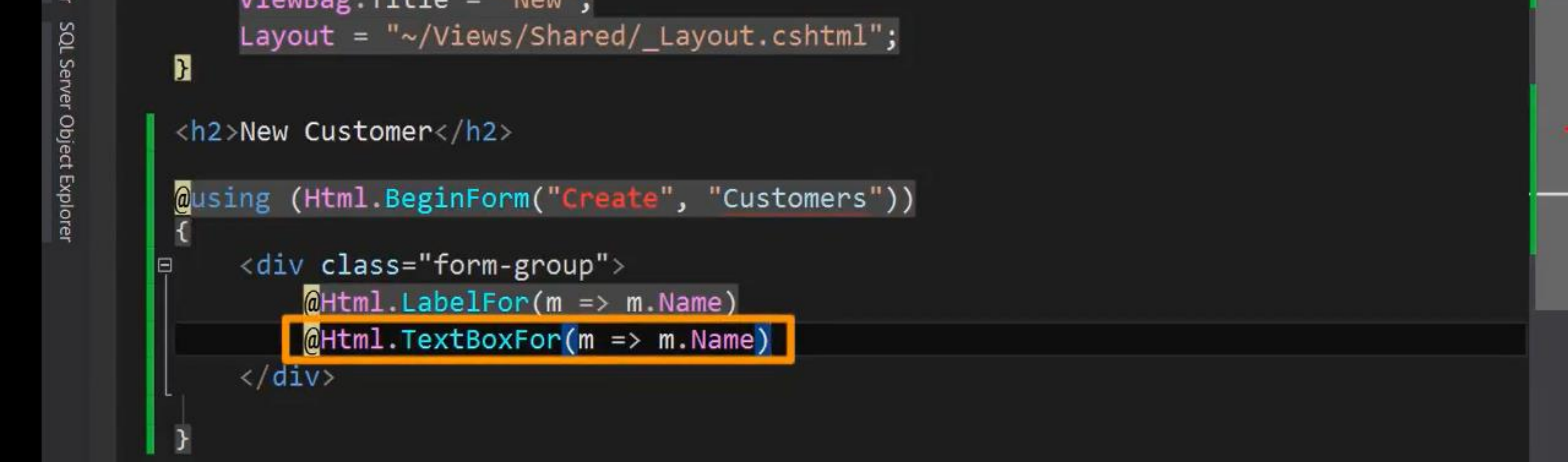


```
▼<form action="/Customers/Create" method="post">
  ▼<div class="form-group">
      <label for="Name">Name</label>
      <input data-val="true" data-val-length="The field Name
      must be a string with a maximum length of 255." data-val-
      length-max="255" data-val-required="The Name field is
      required." id="Name" name="Name" type="text" value>
    </div>
  </form>
  <hr>
```

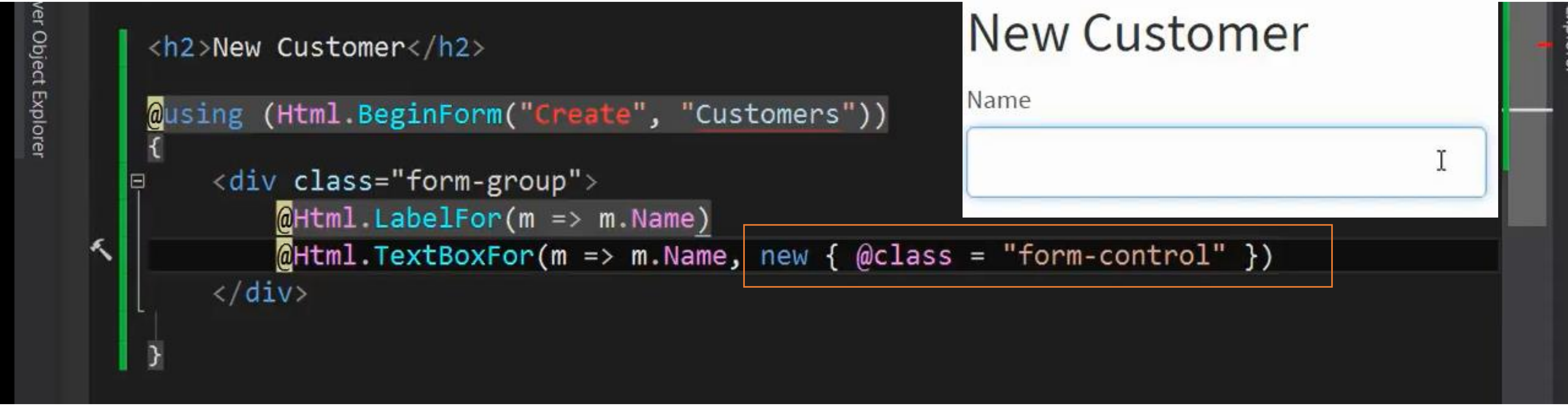
```
element.style {
}
input, select, textarea {
   max-width: 280px;
}
input, button, select,        bootst
```



```
{
    public class Customer
    {
        public int Id { get; set; }
        [Required]
        [StringLength(255)]
        public string Name { get; set; }
```



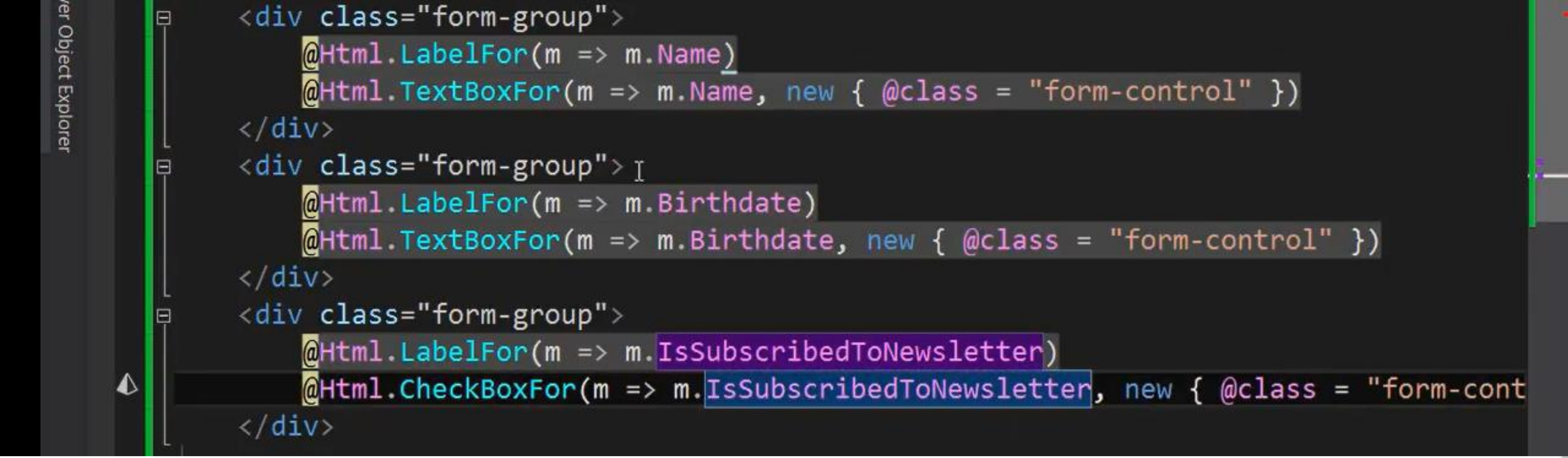
```
    ViewBag.Title = "New";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<h2>New Customer</h2>

@using (Html.BeginForm("Create", "Customers"))
{
    <div class="form-group">
        @Html.LabelFor(m => m.Name)
        @Html.TextBoxFor(m => m.Name)
    </div>
}
```



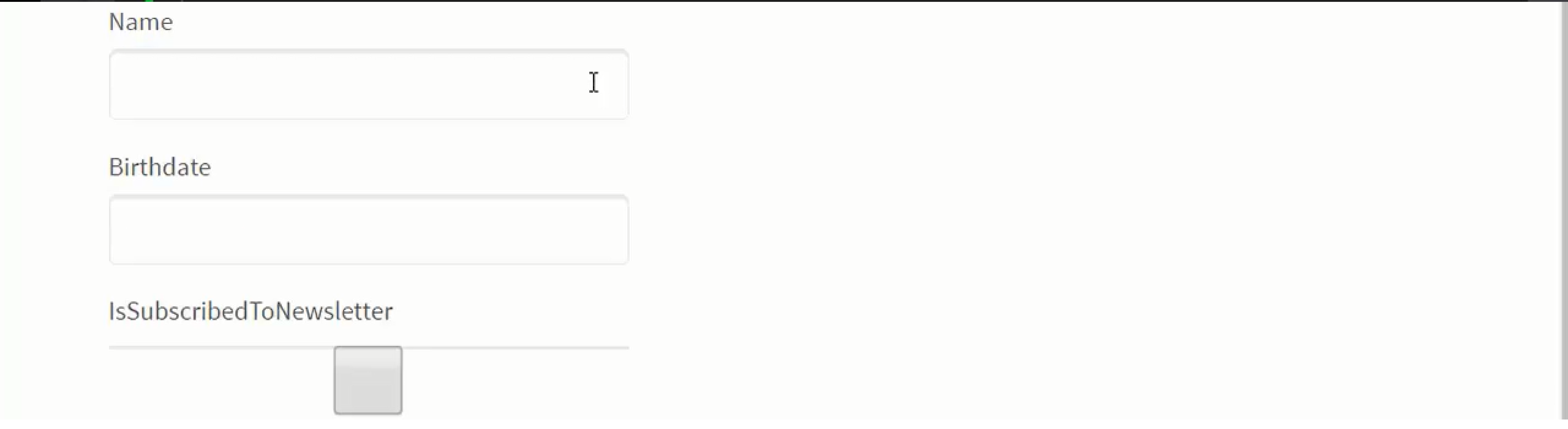
```
<h2>New Customer</h2>

@using (Html.BeginForm("Create", "Customers"))
{
    <div class="form-group">
        @Html.LabelFor(m => m.Name)
        @Html.TextBoxFor(m => m.Name, new { @class = "form-control" })
    </div>
}
```

New Customer

Name



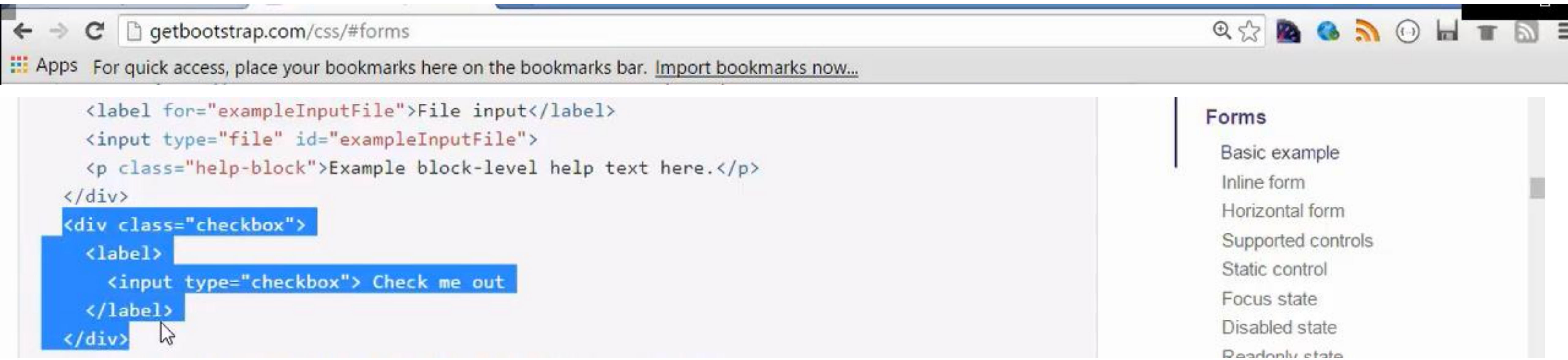
```
<div class="form-group">
    @Html.LabelFor(m => m.Name)
    @Html.TextBoxFor(m => m.Name, new { @class = "form-control" })
</div>
<div class="form-group">
    @Html.LabelFor(m => m.Birthdate)
    @Html.TextBoxFor(m => m.Birthdate, new { @class = "form-control" })
</div>
<div class="form-group">
    @Html.LabelFor(m => m.IsSubscribedToNewsletter)
    @Html.CheckBoxFor(m => m.IsSubscribedToNewsletter, new { @class = "form-cont
</div>
```



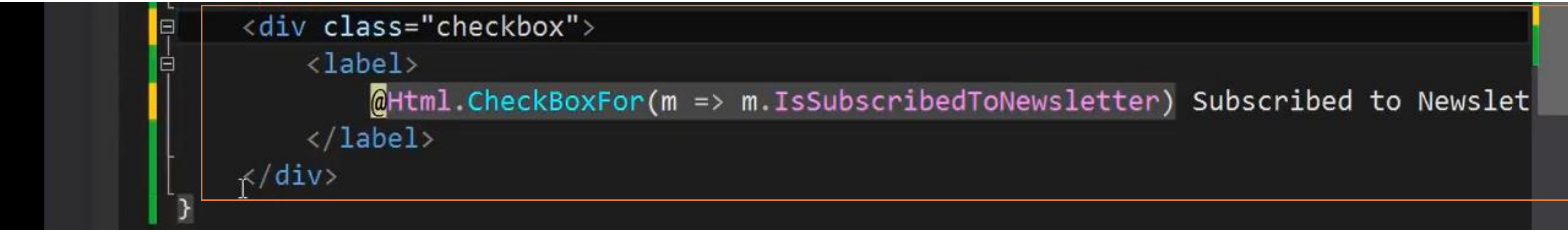
Name

Birthdate

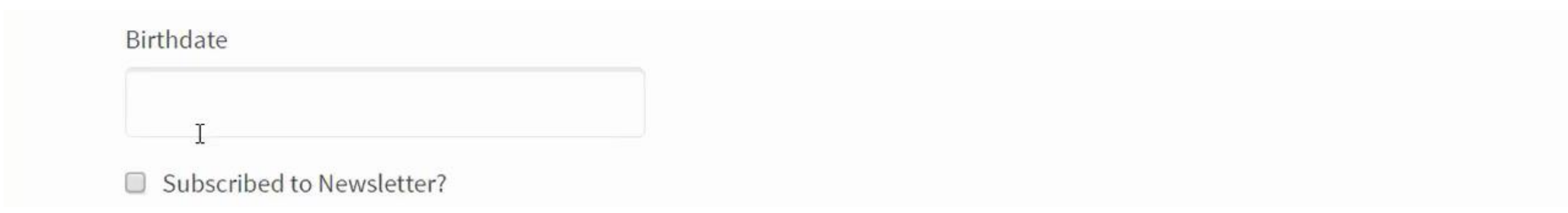
IsSubscribedToNewsletter



```
    <label for="exampleInputFile">File input</label>
    <input type="file" id="exampleInputFile">
    <p class="help-block">Example block-level help text here.</p>
  </div>
  <div class="checkbox">
    <label>
      <input type="checkbox"> Check me out
    </label>
  </div>
```



```
    <div class="checkbox">
        <label>
            @Html.CheckBoxFor(m => m.IsSubscribedToNewsletter) Subscribed to Newslet
        </label>
    </div>
}
```



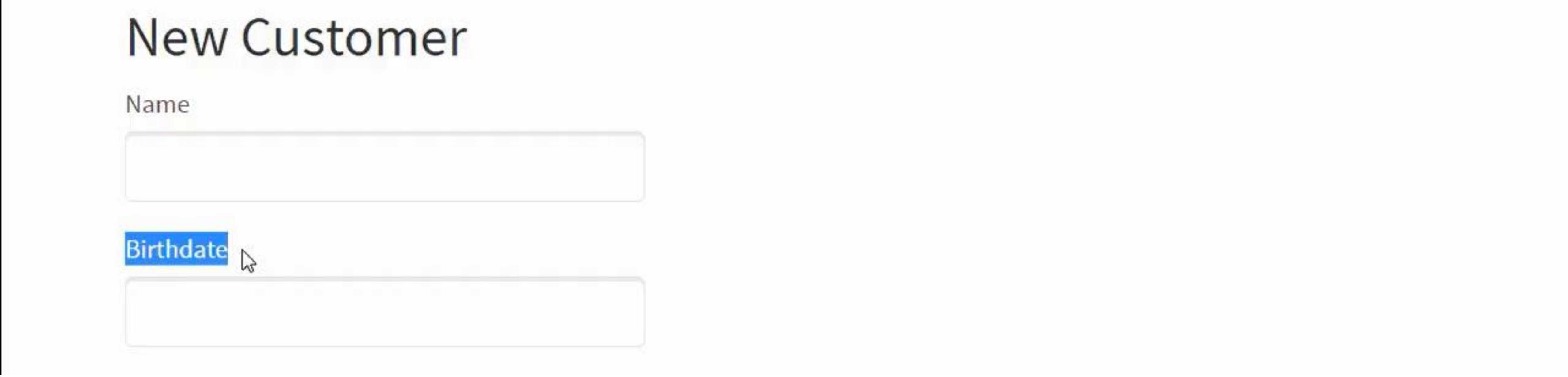
Birthdate

□ Subscribed to Newsletter?

Form Labels

Zouhair Rimale, Ph.D.

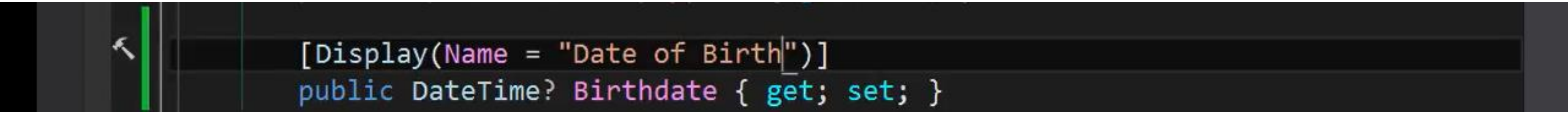
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



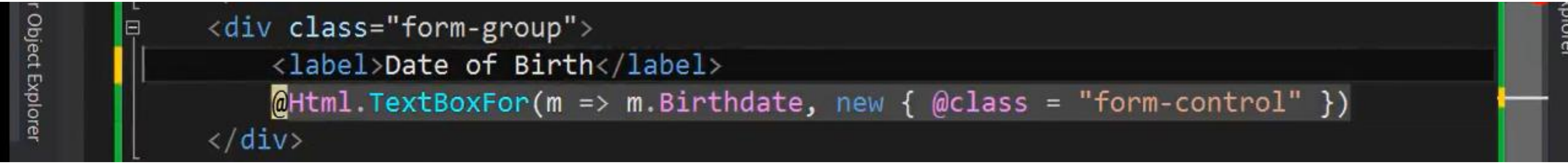
New Customer

Name

Birthdate



```
[Display(Name = "Date of Birth")]
public DateTime? Birthdate { get; set; }
```

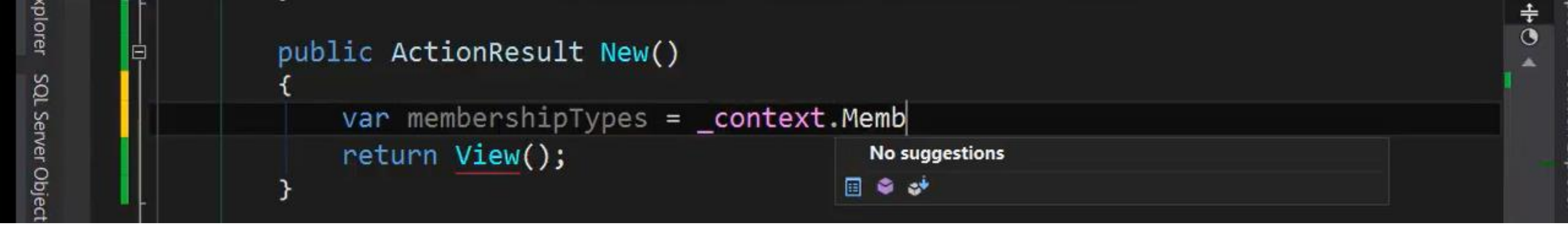


```
<div class="form-group">
    <label>Date of Birth</label>
    @Html.TextBoxFor(m => m.Birthdate, new { @class = "form-control" })
</div>
```

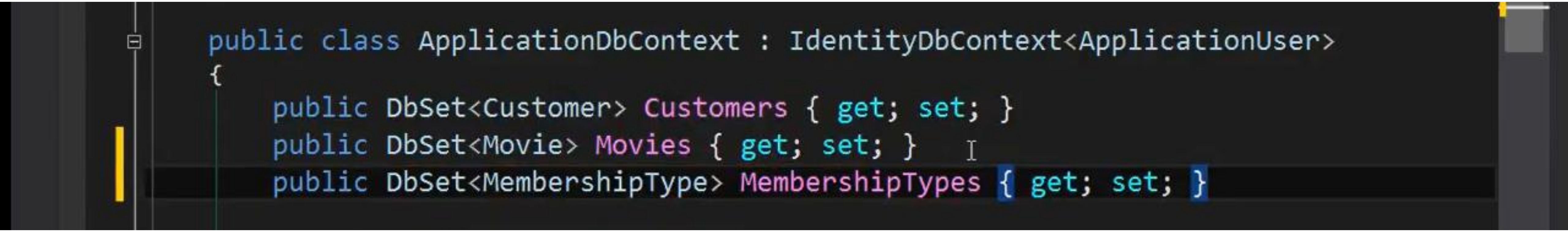
Drop-down Lists

Zouhair Rimale, Ph.D.

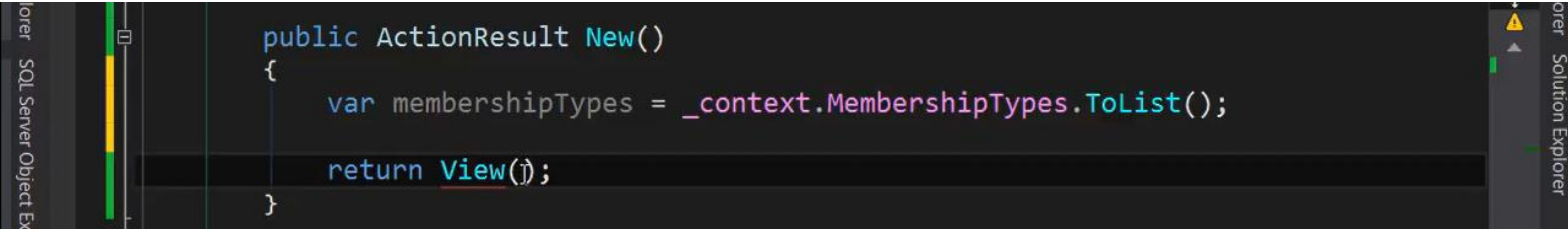
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



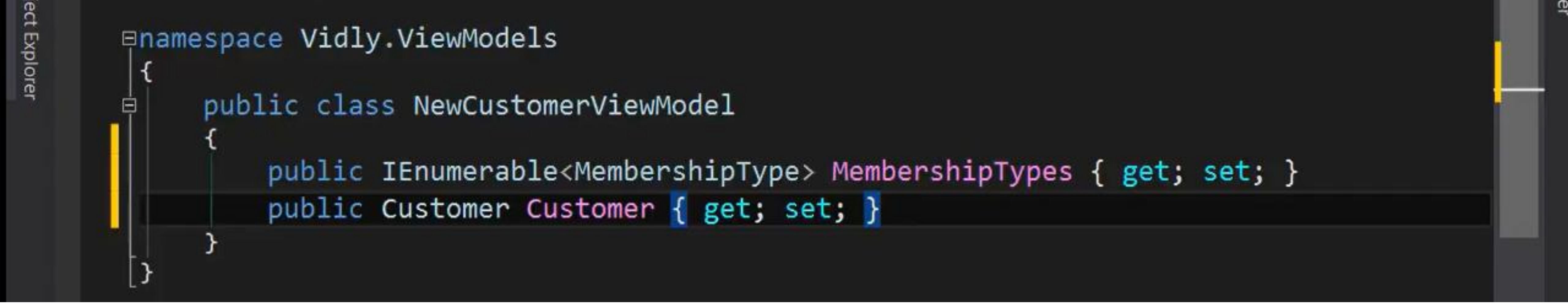
```
public ActionResult New()
{
    var membershipTypes = _context.Memb
    return View();
}
```



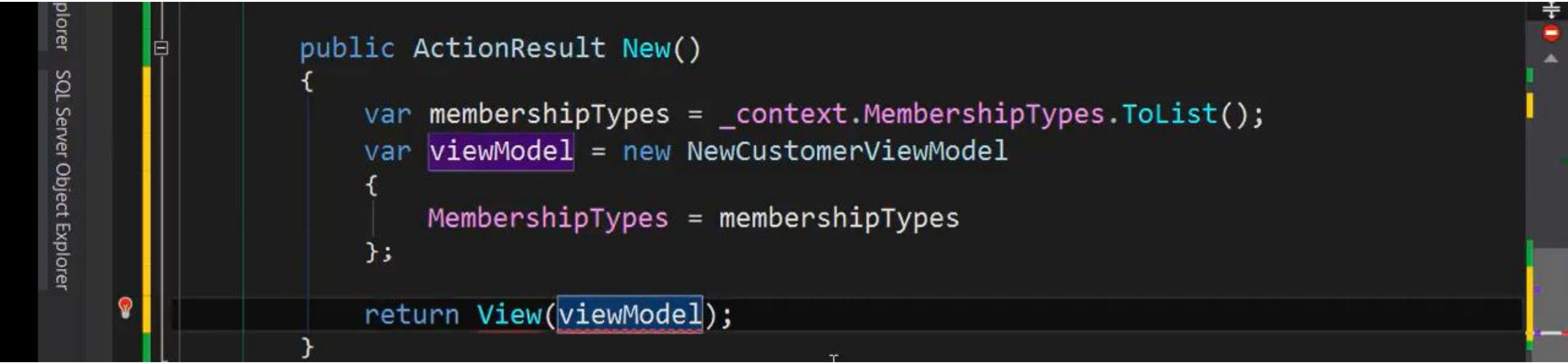
```
public class ApplicationDbContext : IdentityDbContext<ApplicationUser>
{
    public DbSet<Customer> Customers { get; set; }
    public DbSet<Movie> Movies { get; set; }
    public DbSet<MembershipType> MembershipTypes { get; set; }
```



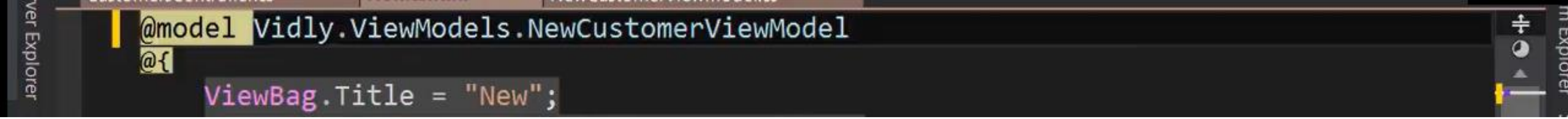
```
public ActionResult New()
{
    var membershipTypes = _context.MembershipTypes.ToList();

    return View();
}
```



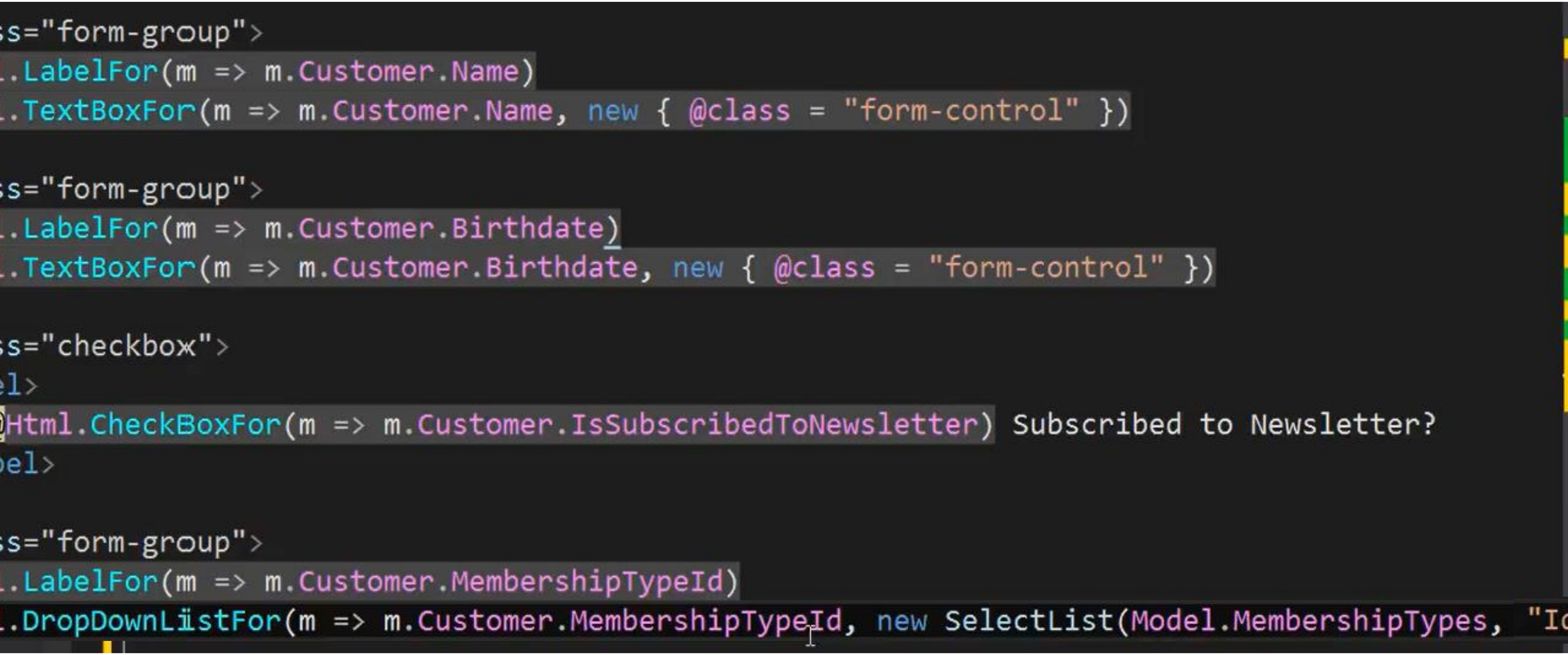
```
namespace Vidly.ViewModels
{
    public class NewCustomerViewModel
    {
        public IEnumerable<MembershipType> MembershipTypes { get; set; }
        public Customer Customer { get; set; }
    }
}
```



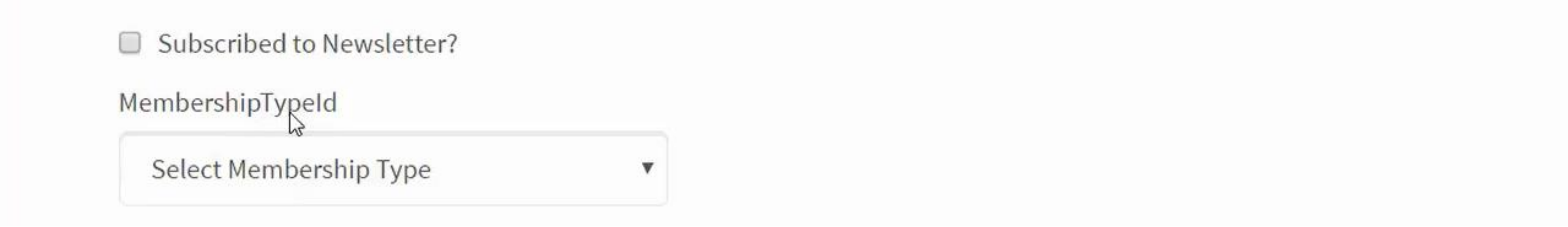
```
        public ActionResult New()
        {
            var membershipTypes = _context.MembershipTypes.ToList();
            var viewModel = new NewCustomerViewModel
            {
                MembershipTypes = membershipTypes
            };

            return View(viewModel);
        }
```



```
@model Vidly.ViewModels.NewCustomerViewModel
@{
    ViewBag.Title = "New";
```



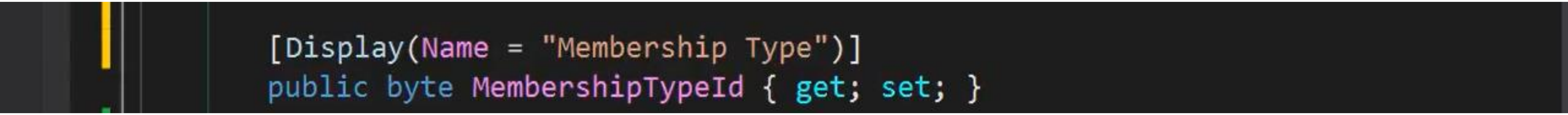
```
ss="form-group">
l.LabelFor(m => m.Customer.Name)
l.TextBoxFor(m => m.Customer.Name, new { @class = "form-control" })

ss="form-group">
l.LabelFor(m => m.Customer.Birthdate)
l.TextBoxFor(m => m.Customer.Birthdate, new { @class = "form-control" })

ss="checkbox">
el>
@Html.CheckBoxFor(m => m.Customer.IsSubscribedToNewsletter) Subscribed to Newsletter?
bel>

ss="form-group">
l.LabelFor(m => m.Customer.MembershipTypeId)
l.DropDownListFor(m => m.Customer.MembershipTypeId, new SelectList(Model.MembershipTypes, "Id
```



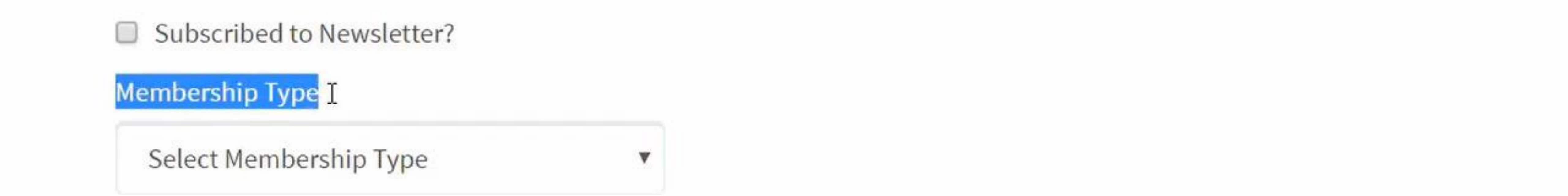
☐ Subscribed to Newsletter?

MembershipTypeId

Select Membership Type ▾



```
[Display(Name = "Membership Type")]
public byte MembershipTypeId { get; set; }
```



☐ Subscribed to Newsletter?

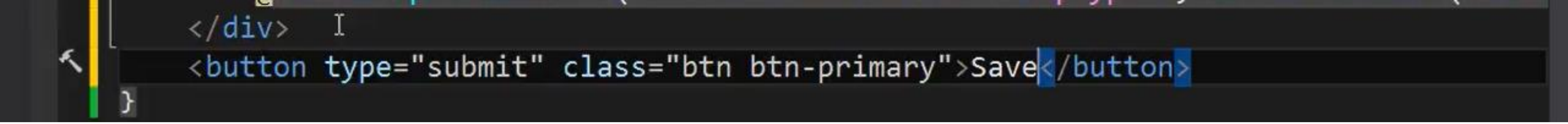
Membership Type

Select Membership Type ▾

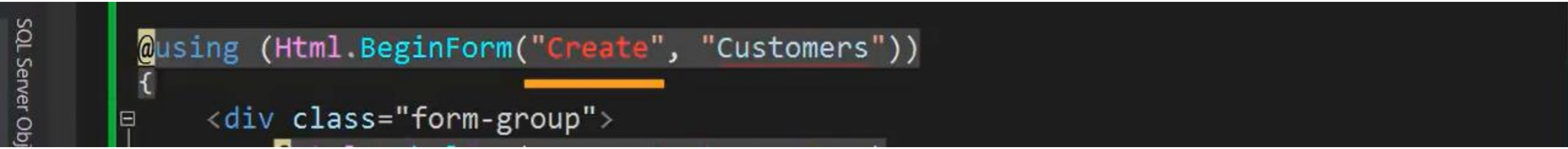
Model Binding

Zouhair Rimale, Ph.D.

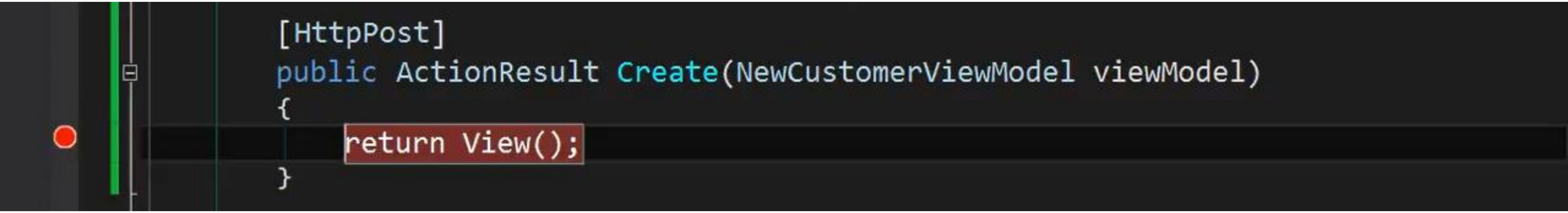
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



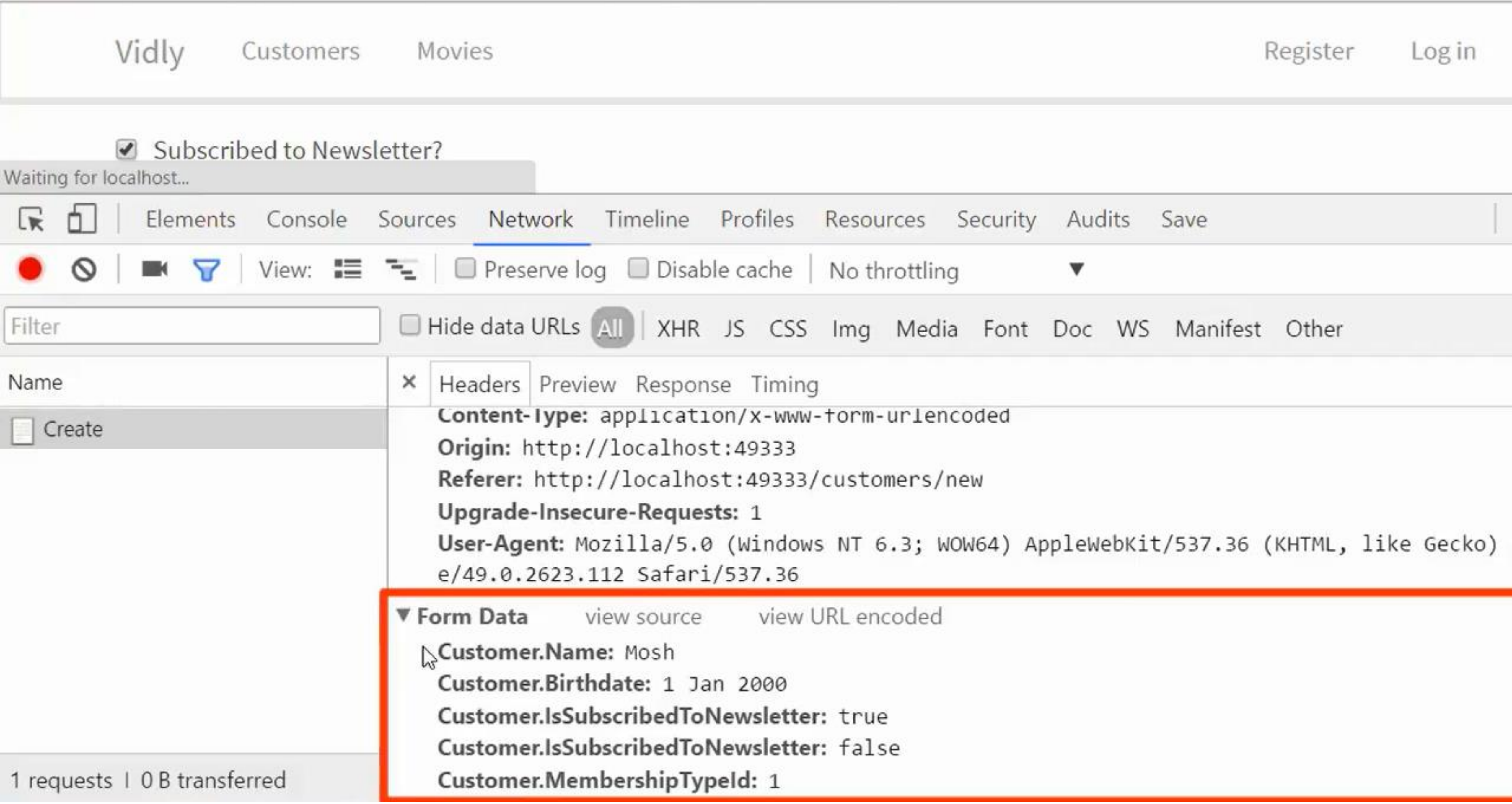
```
    </div>
    <button type="submit" class="btn btn-primary">Save</button>
}
```



```
@using (Html.BeginForm("Create", "Customers"))
{
    <div class="form-group">
```



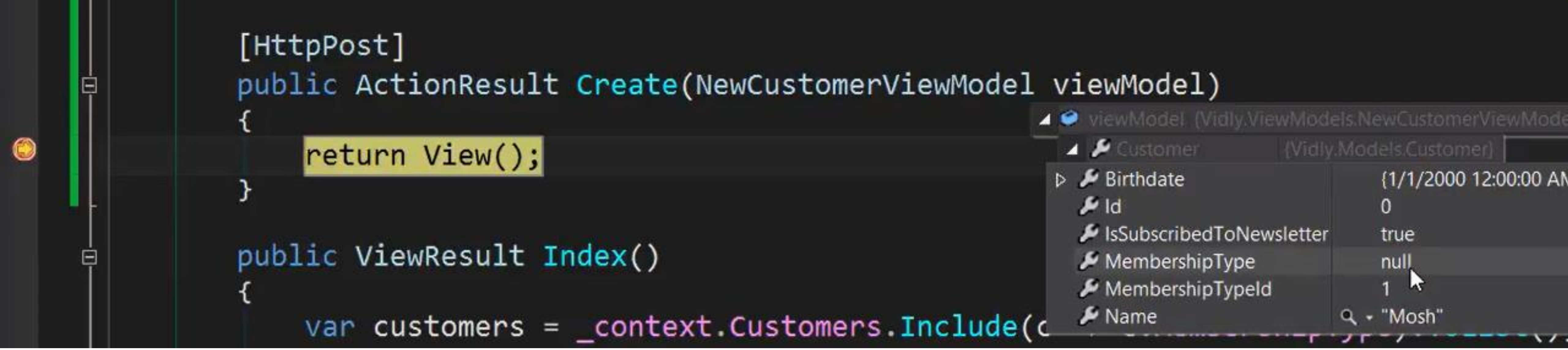
```
[HttpPost]
public ActionResult Create(NewCustomerViewModel viewModel)
{
    return View();
}
```



Content-Type: application/x-www-form-urlencoded
Origin: http://localhost:49333
Referer: http://localhost:49333/customers/new
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)
e/49.0.2623.112 Safari/537.36

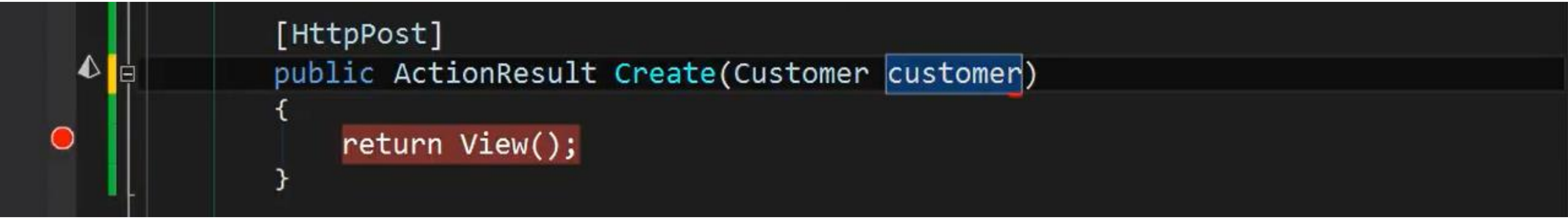
▼ **Form Data** view source view URL encoded

Customer.Name: Mosh
Customer.Birthdate: 1 Jan 2000
Customer.IsSubscribedToNewsletter: true
Customer.IsSubscribedToNewsletter: false
Customer.MembershipTypeId: 1

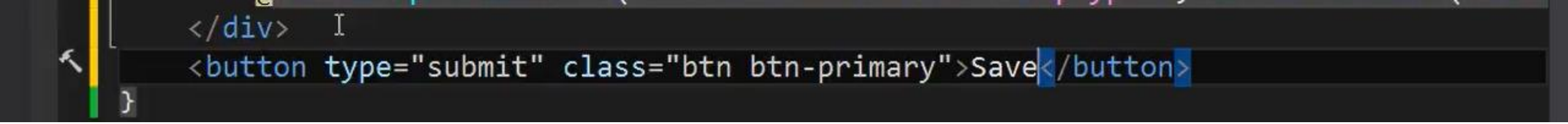


```
[HttpPost]
public ActionResult Create(NewCustomerViewModel viewModel)
{
    return View();
}

public ViewResult Index()
{
    var customers = _context.Customers.Include(c
```

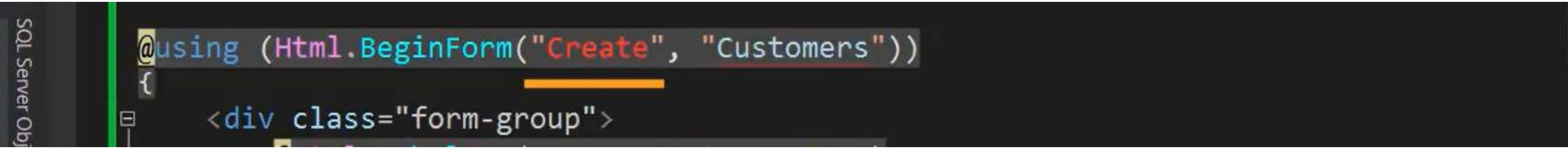
```
◢ viewModel  {Vidly.ViewModels.NewCustomerViewMode
  ◢ Customer                  {Vidly.Models.Customer}
  ▷ Birthdate                 {1/1/2000 12:00:00 AM
    Id                        0
    IsSubscribedToNewsletter  true
    MembershipType            null
    MembershipTypeId          1
    Name                      "Mosh"
```



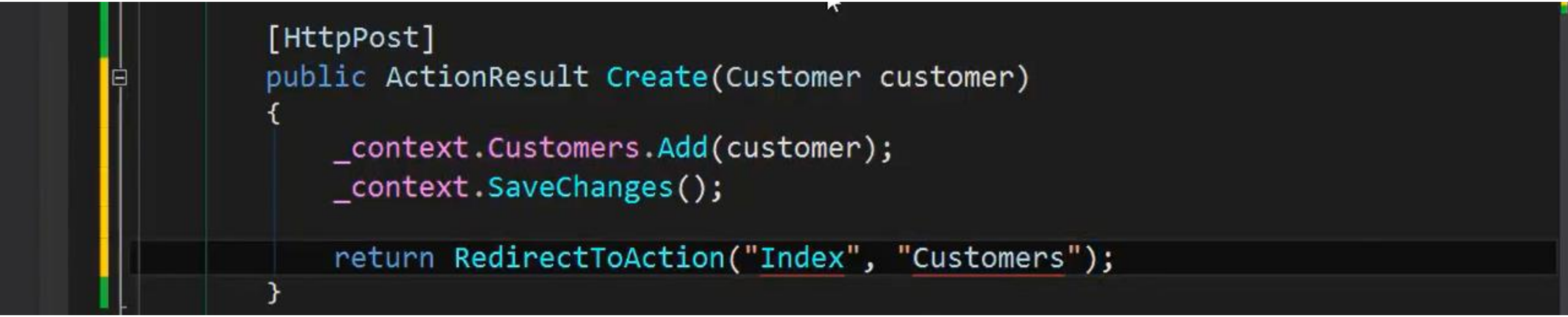
```
[HttpPost]
public ActionResult Create(Customer customer)
{
    return View();
}
```



```
    </div>
    <button type="submit" class="btn btn-primary">Save</button>
}
```



```
@using (Html.BeginForm("Create", "Customers"))
{
    <div class="form-group">
```

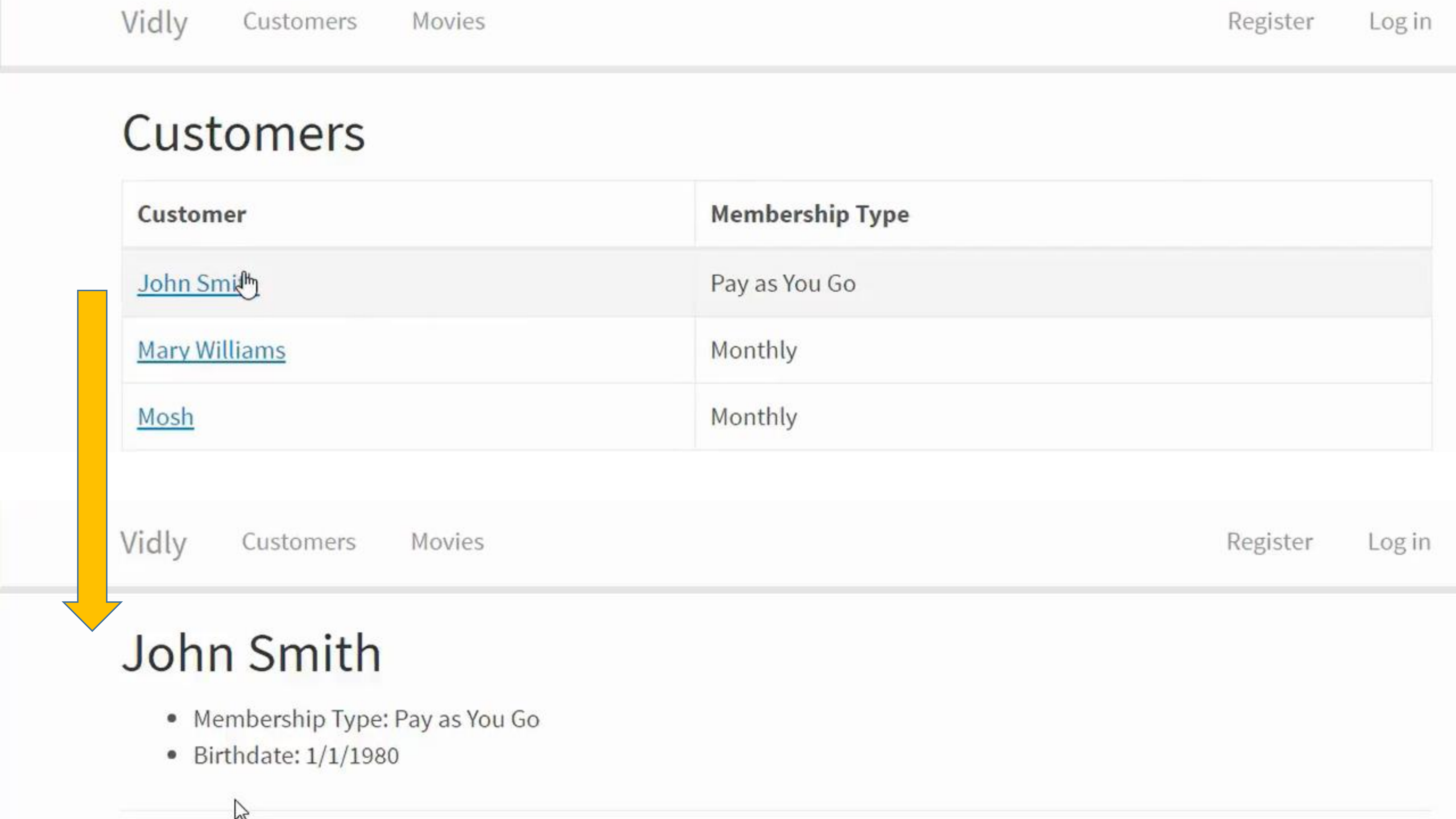


```
[HttpPost]
public ActionResult Create(Customer customer)
{
    _context.Customers.Add(customer);
    _context.SaveChanges();

    return RedirectToAction("Index", "Customers");
}
```

Edit Form

Zouhair Rimale, Ph.D.

Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular

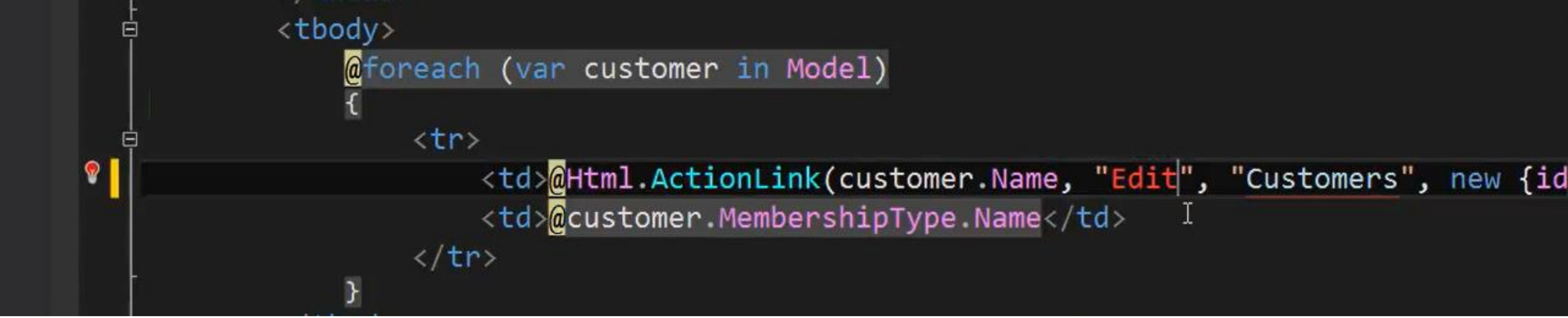


Customers

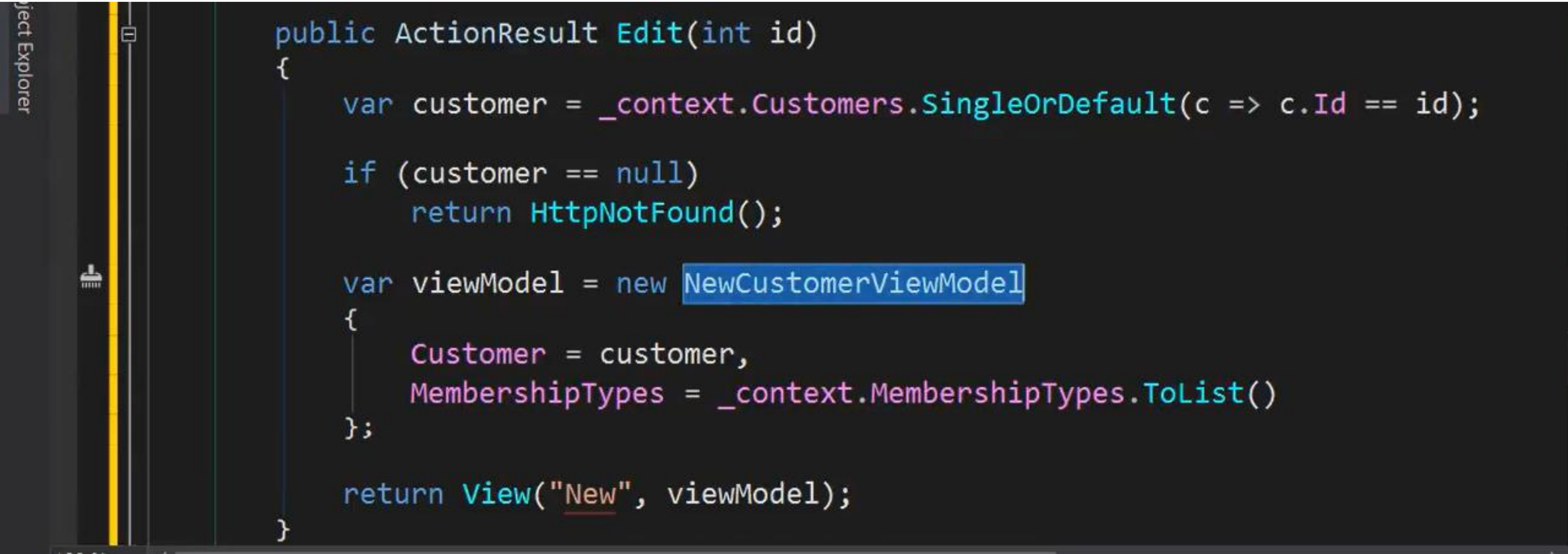
Customer	Membership Type
John Smith	Pay as You Go
Mary Williams	Monthly
Mosh	Monthly

John Smith

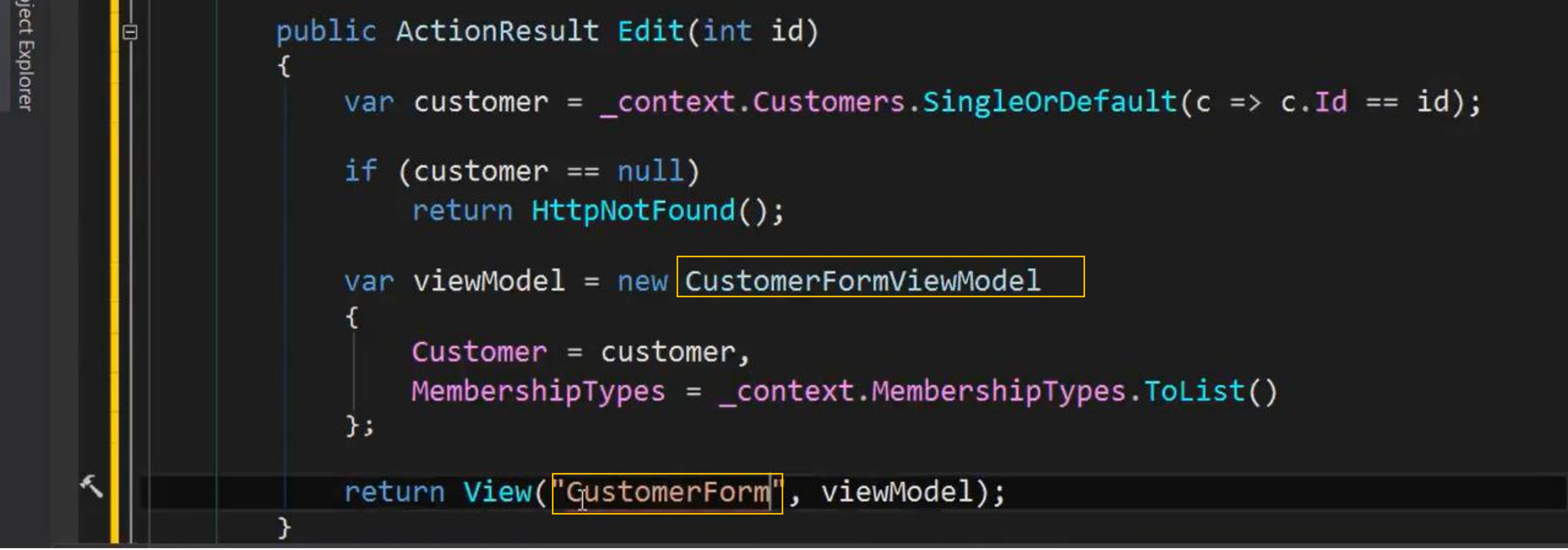
- Membership Type: Pay as You Go
- Birthdate: 1/1/1980



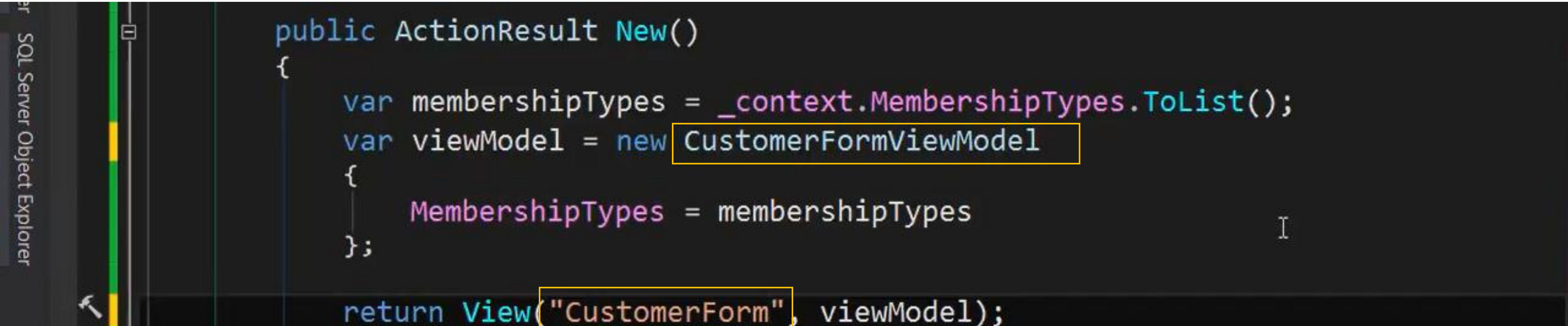
```
<tbody>
    @foreach (var customer in Model)
    {
        <tr>
            <td>@Html.ActionLink(customer.Name, "Edit", "Customers", new {id
            <td>@customer.MembershipType.Name</td>
        </tr>
    }
```



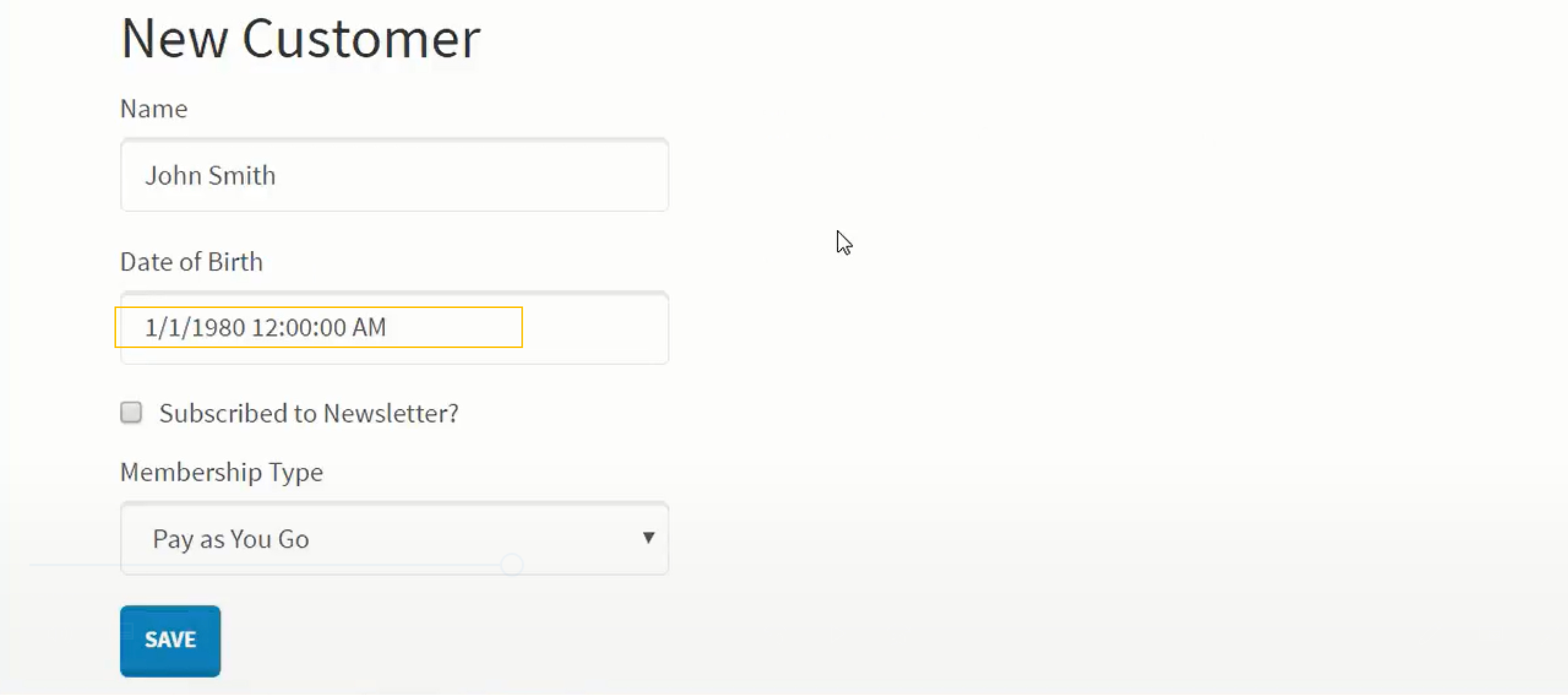
```
public ActionResult Edit(int id)
{
    var customer = _context.Customers.SingleOrDefault(c => c.Id == id);

    if (customer == null)
        return HttpNotFound();

    var viewModel = new NewCustomerViewModel
    {
        Customer = customer,
        MembershipTypes = _context.MembershipTypes.ToList()
    };

    return View("New", viewModel);
}
```



```
public ActionResult Edit(int id)
{
    var customer = _context.Customers.SingleOrDefault(c => c.Id == id);

    if (customer == null)
        return HttpNotFound();

    var viewModel = new CustomerFormViewModel
    {
        Customer = customer,
        MembershipTypes = _context.MembershipTypes.ToList()
    };

    return View("CustomerForm", viewModel);
}
```



```
public ActionResult New()
{
    var membershipTypes = _context.MembershipTypes.ToList();
    var viewModel = new CustomerFormViewModel
    {
        MembershipTypes = membershipTypes
    };

    return View("CustomerForm", viewModel);
```



New Customer

Name

John Smith

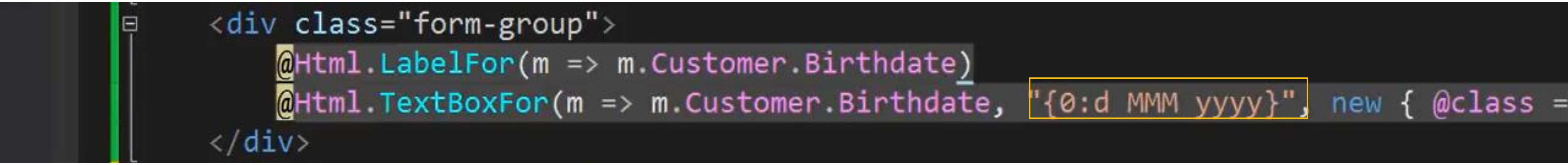
Date of Birth

1/1/1980 12:00:00 AM

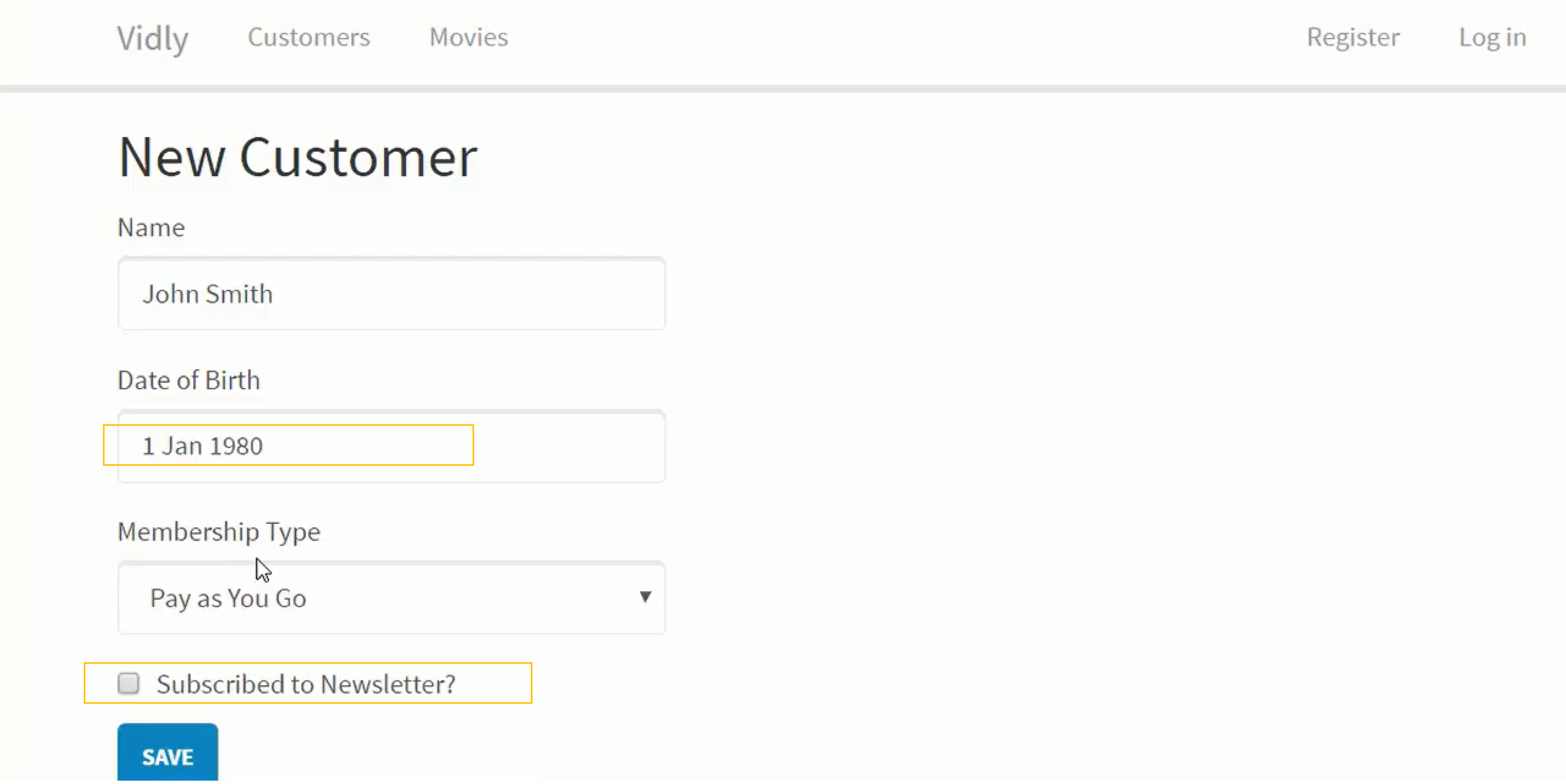
☐ Subscribed to Newsletter?

Membership Type

Pay as You Go ▾



```
    <div class="form-group">
        @Html.LabelFor(m => m.Customer.Birthdate)
        @Html.TextBoxFor(m => m.Customer.Birthdate, "{0:d MMM yyyy}", new { @class =
    </div>
```



New Customer

Name

John Smith

Date of Birth

1 Jan 1980

Membership Type

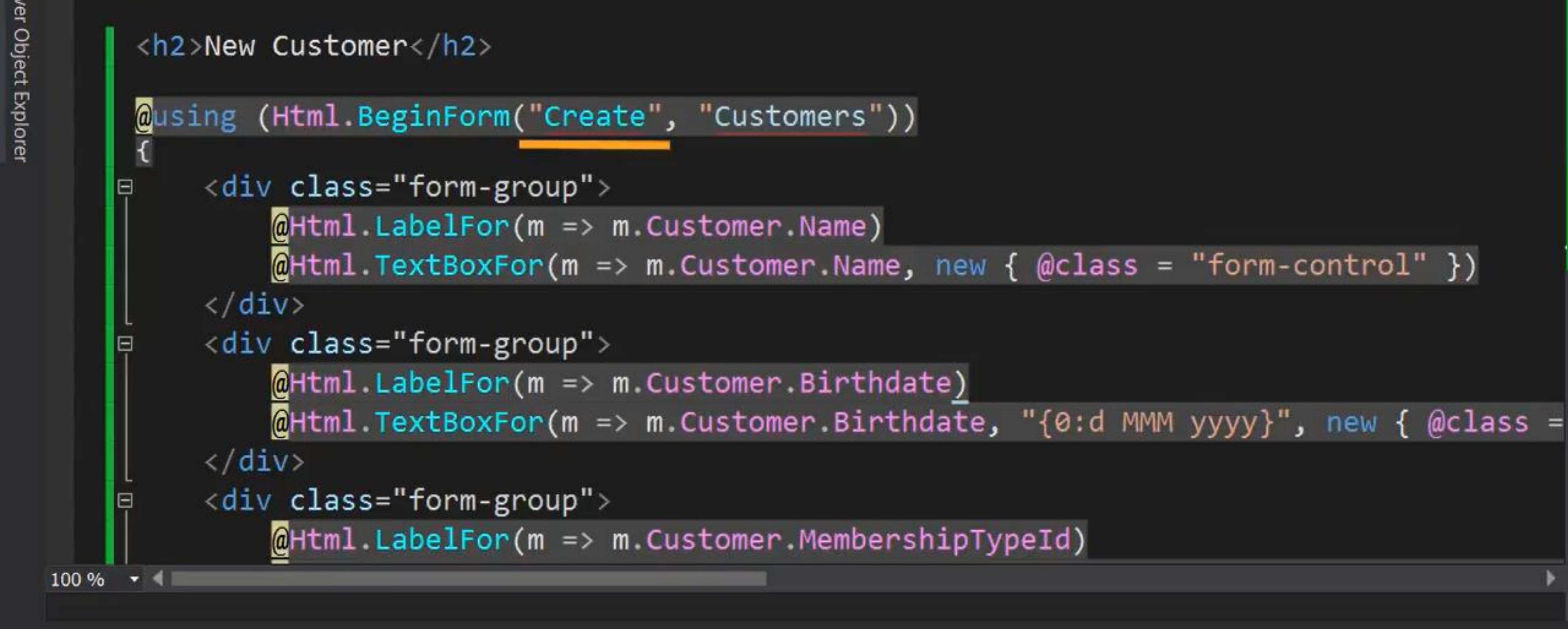
Pay as You Go ▼

□ Subscribed to Newsletter?

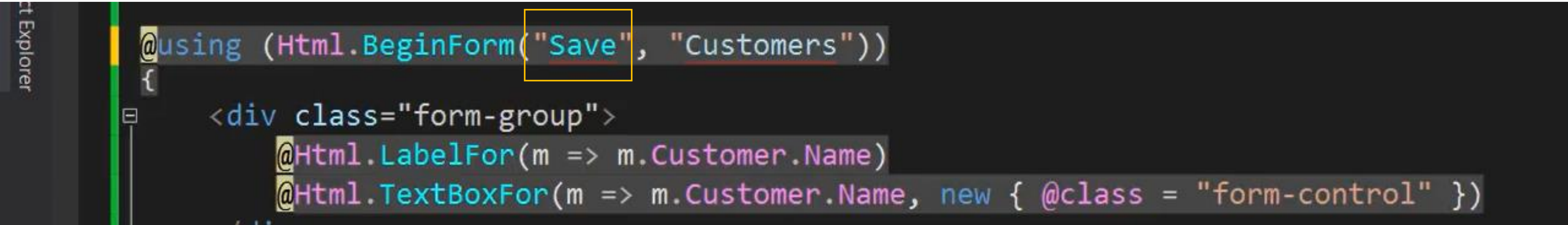
Updating Data

Zouhair Rimale, Ph.D.

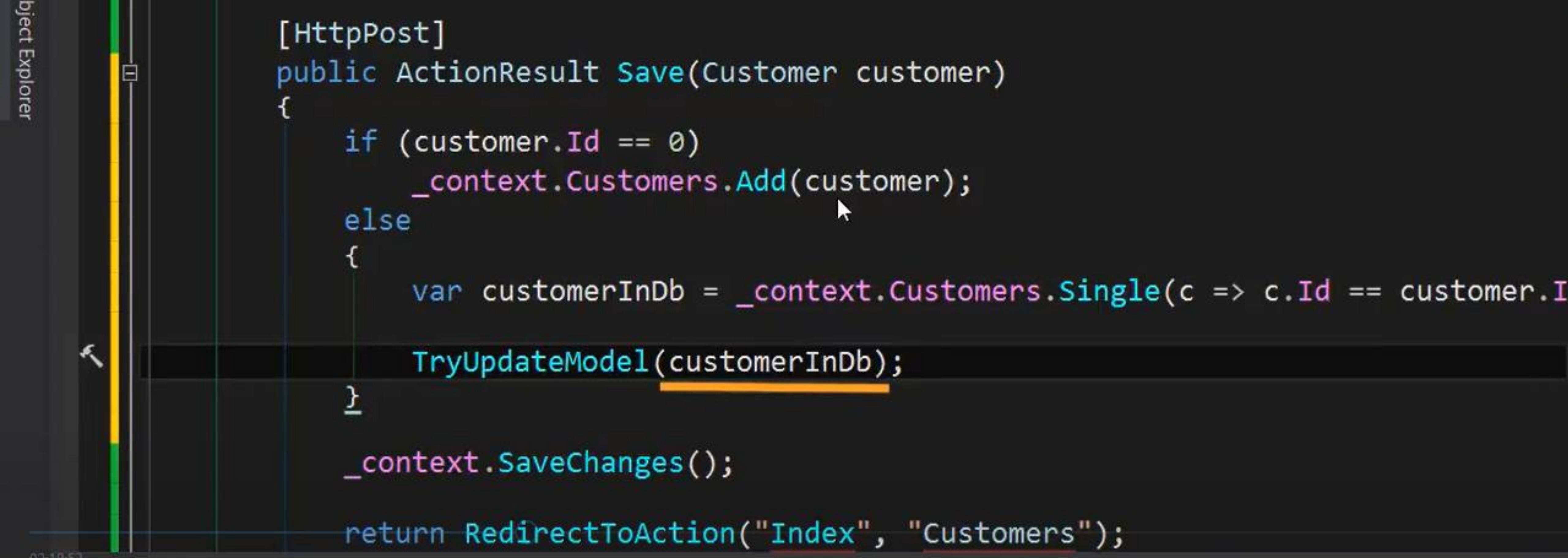
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



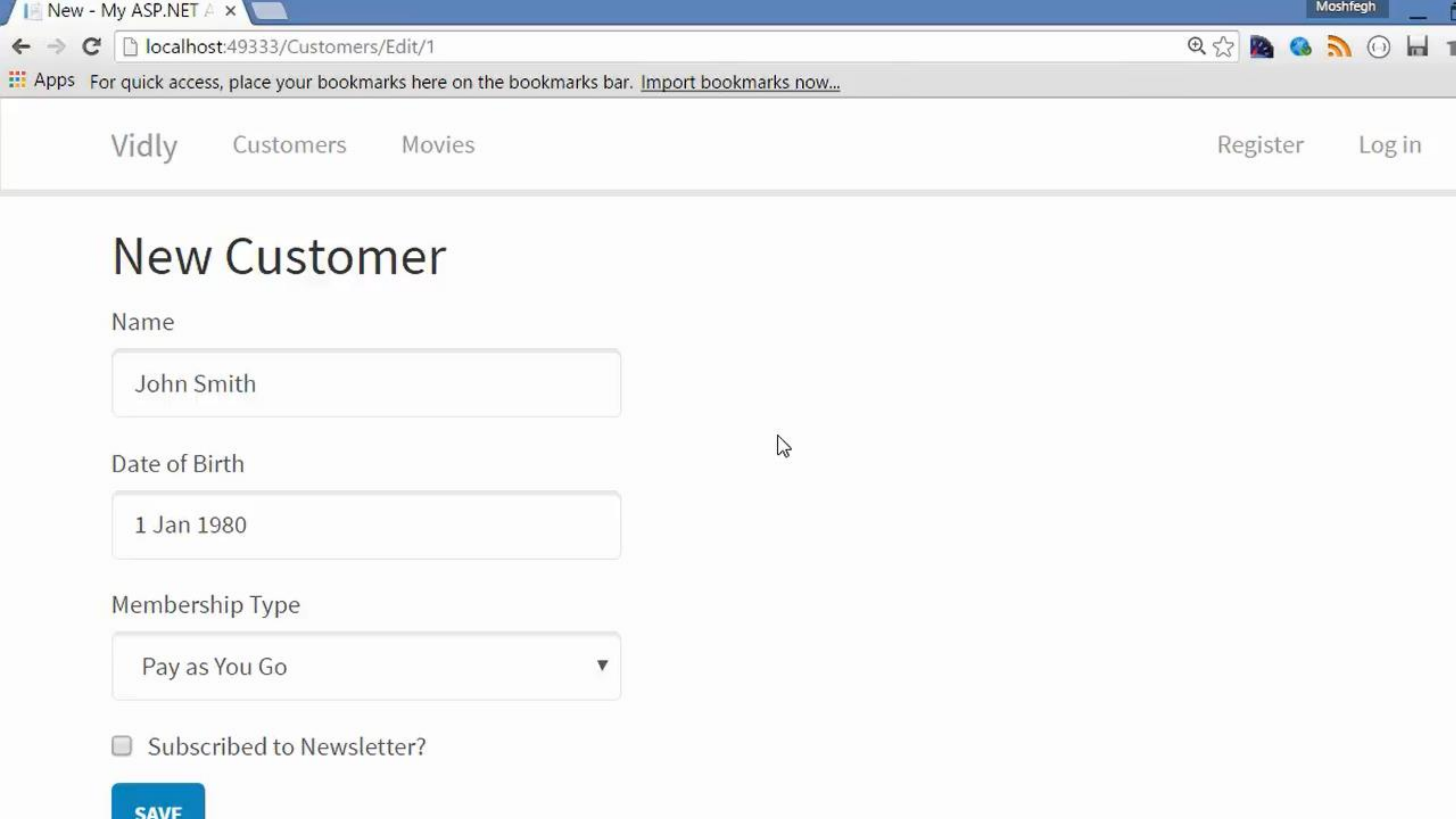
```
<h2>New Customer</h2>

@using (Html.BeginForm("Create", "Customers"))
{
    <div class="form-group">
        @Html.LabelFor(m => m.Customer.Name)
        @Html.TextBoxFor(m => m.Customer.Name, new { @class = "form-control" })
    </div>
    <div class="form-group">
        @Html.LabelFor(m => m.Customer.Birthdate)
        @Html.TextBoxFor(m => m.Customer.Birthdate, "{0:d MMM yyyy}", new { @class =
    </div>
    <div class="form-group">
        @Html.LabelFor(m => m.Customer.MembershipTypeId)
```



```
@using (Html.BeginForm("Save", "Customers"))
{
    <div class="form-group">
        @Html.LabelFor(m => m.Customer.Name)
        @Html.TextBoxFor(m => m.Customer.Name, new { @class = "form-control" })
```



```
[HttpPost]
public ActionResult Save(Customer customer)
{
    if (customer.Id == 0)
        _context.Customers.Add(customer);
    else
    {
        var customerInDb = _context.Customers.Single(c => c.Id == customer.I

        TryUpdateModel(customerInDb);
    }

    _context.SaveChanges();

    return RedirectToAction("Index", "Customers");
```


New - My ASP.NET A
Moshfegh
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Vidly
Customers
Movies
Register
Log in

localhost:49333/Customers/Edit/1

New Customer

Name

John Smith

Date of Birth

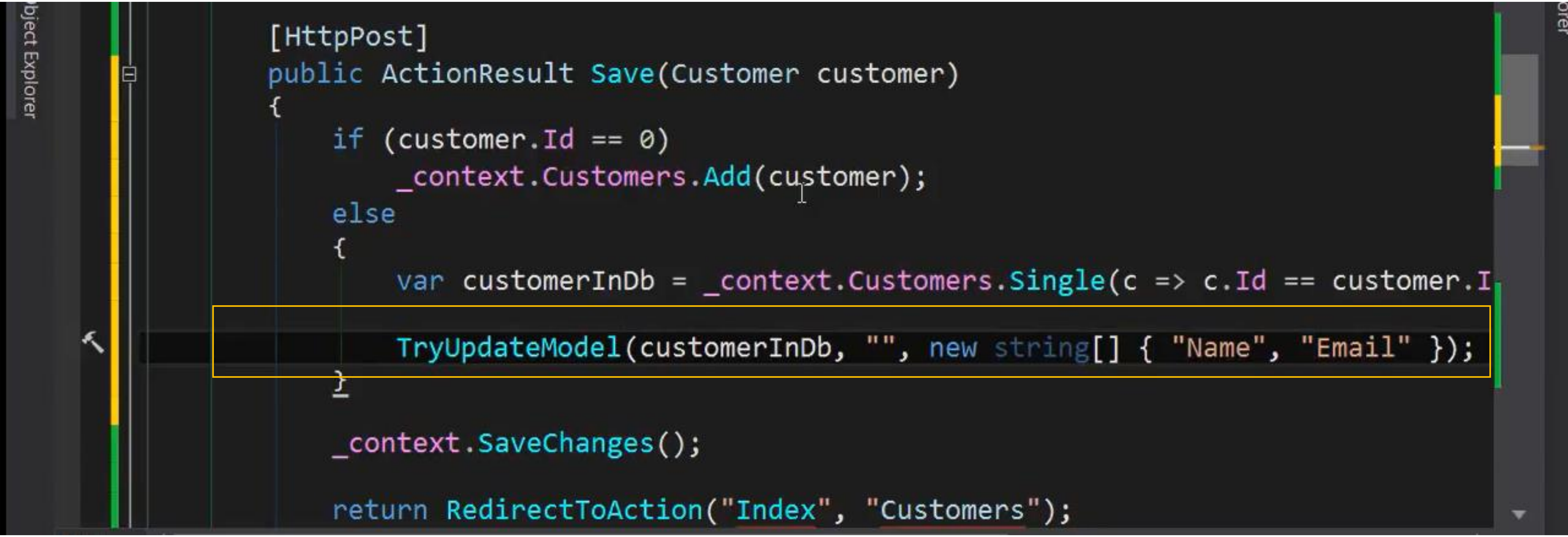
1 Jan 1980

Membership Type

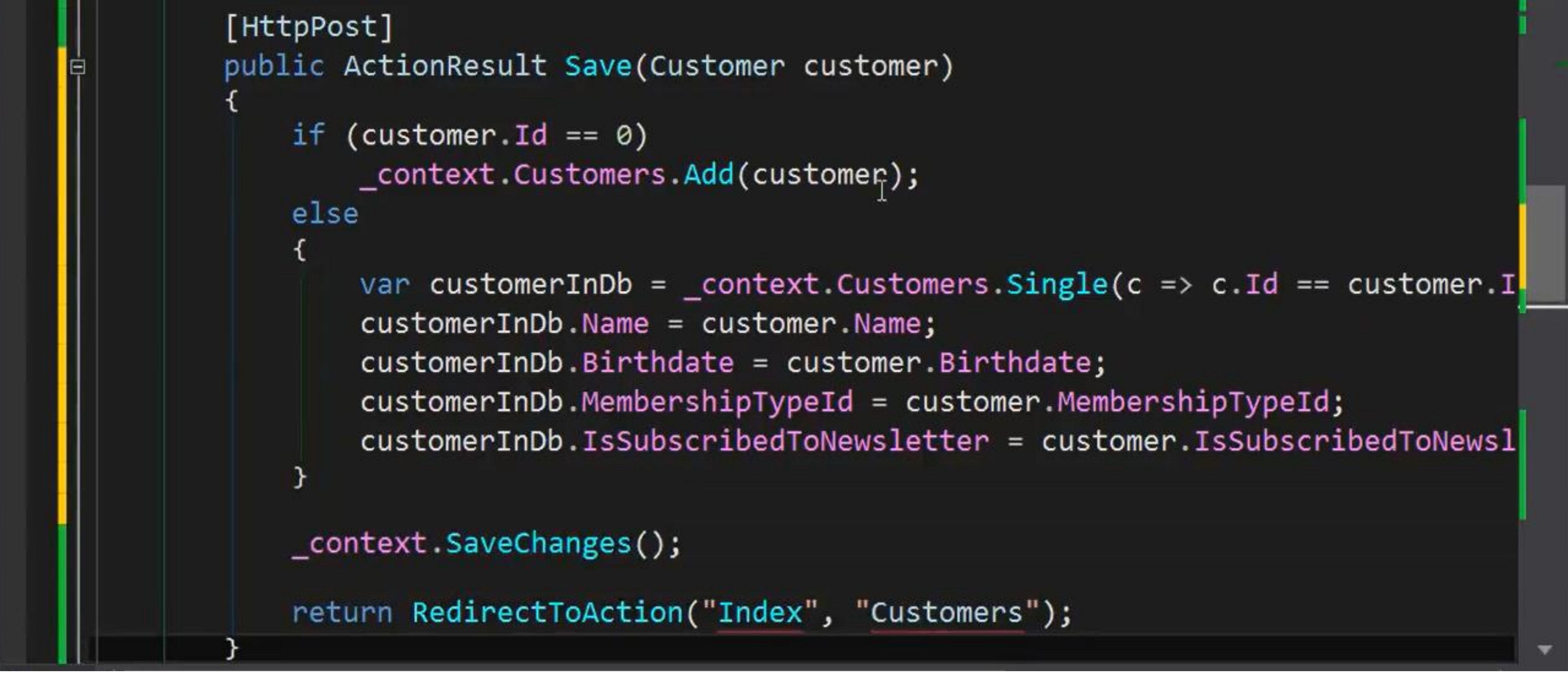
Pay as You Go ▾

□ Subscribed to Newsletter?

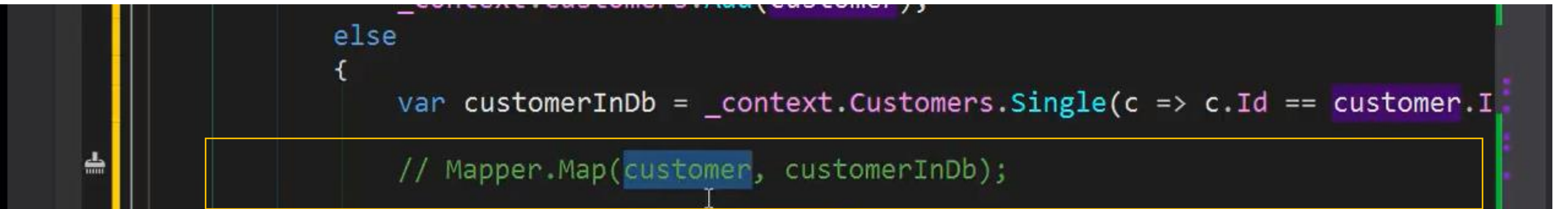
SAVE



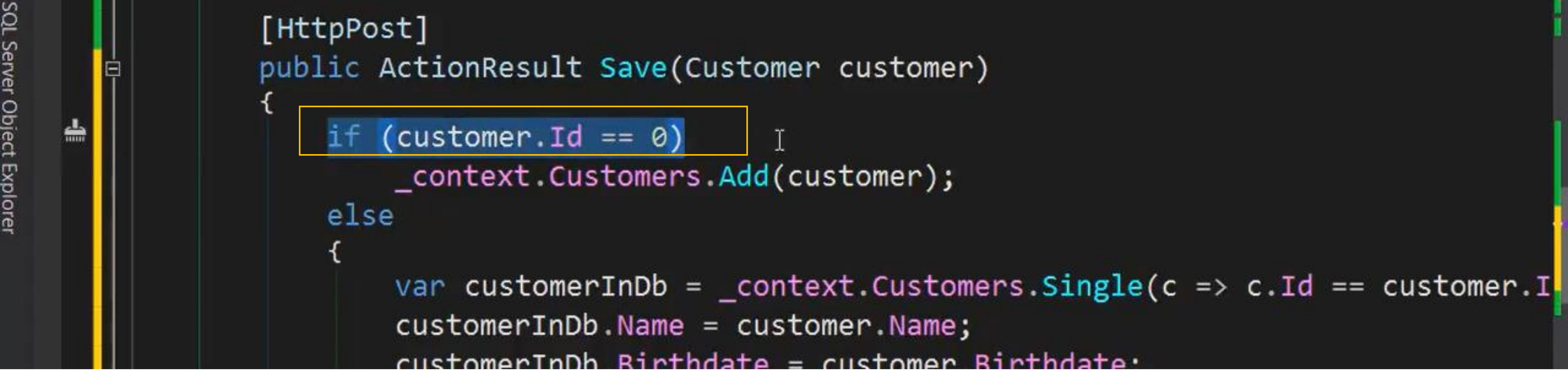
```
[HttpPost]
public ActionResult Save(Customer customer)
{
    if (customer.Id == 0)
        _context.Customers.Add(customer);
    else
    {
        var customerInDb = _context.Customers.Single(c => c.Id == customer.I

        TryUpdateModel(customerInDb, "", new string[] { "Name", "Email" });
    }

    _context.SaveChanges();

    return RedirectToAction("Index", "Customers");
```



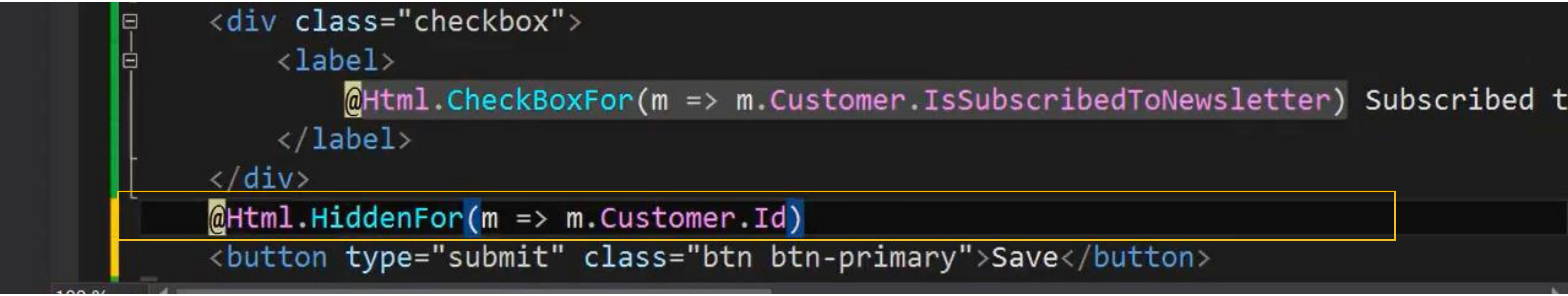
```
[HttpPost]
public ActionResult Save(Customer customer)
{
    if (customer.Id == 0)
        _context.Customers.Add(customer);
    else
    {
        var customerInDb = _context.Customers.Single(c => c.Id == customer.I
        customerInDb.Name = customer.Name;
        customerInDb.Birthdate = customer.Birthdate;
        customerInDb.MembershipTypeId = customer.MembershipTypeId;
        customerInDb.IsSubscribedToNewsletter = customer.IsSubscribedToNewsl
    }

    _context.SaveChanges();

    return RedirectToAction("Index", "Customers");
}
```



```
        _context.Customers.Add(customer);
    else
    {
        var customerInDb = _context.Customers.Single(c => c.Id == customer.I

        // Mapper.Map(customer, customerInDb);
```

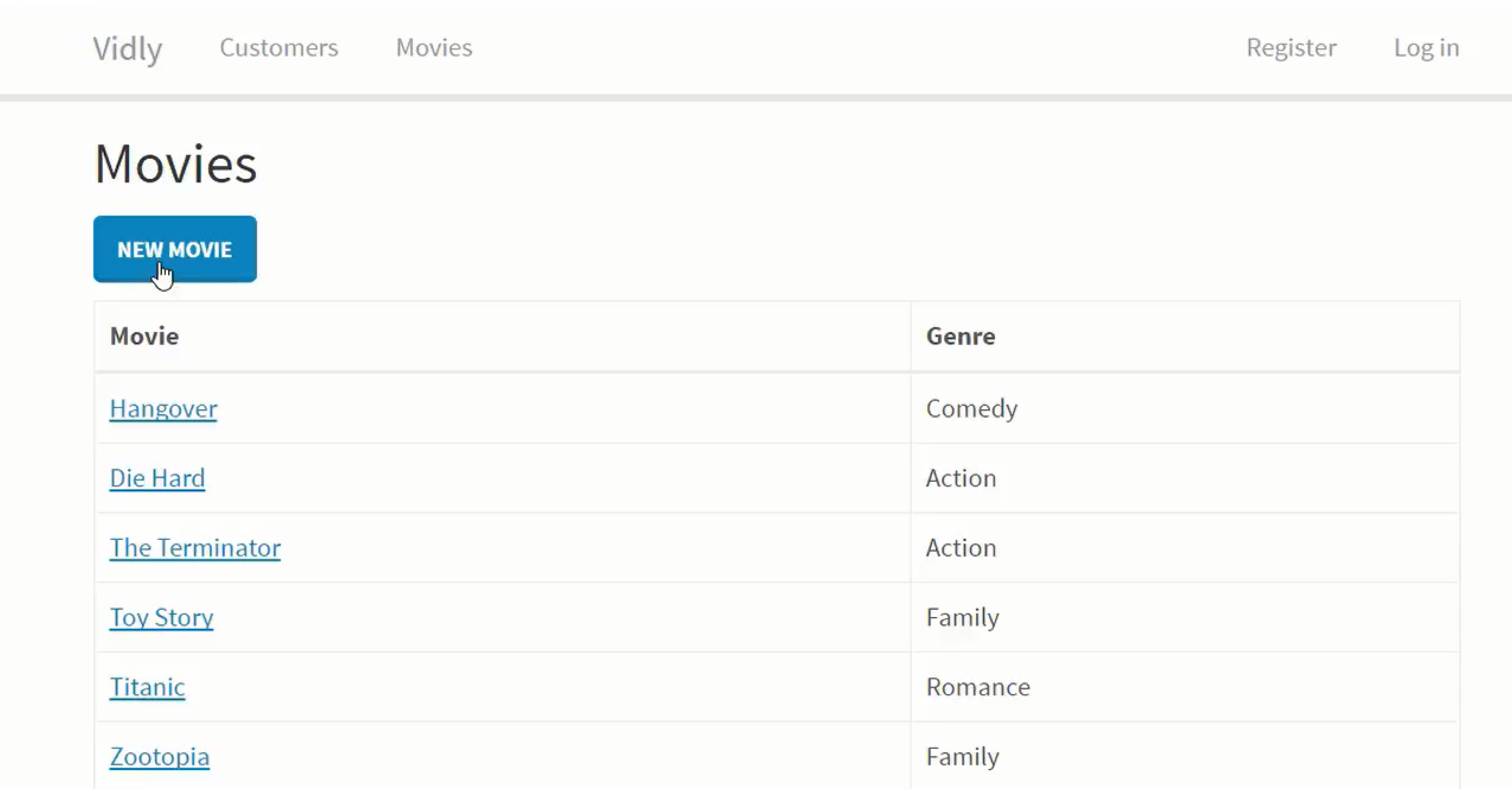


```
[HttpPost]
public ActionResult Save(Customer customer)
{
    if (customer.Id == 0)
        _context.Customers.Add(customer);
    else
    {
        var customerInDb = _context.Customers.Single(c => c.Id == customer.I
        customerInDb.Name = customer.Name;
        customerInDb.Birthdate = customer.Birthdate;
```



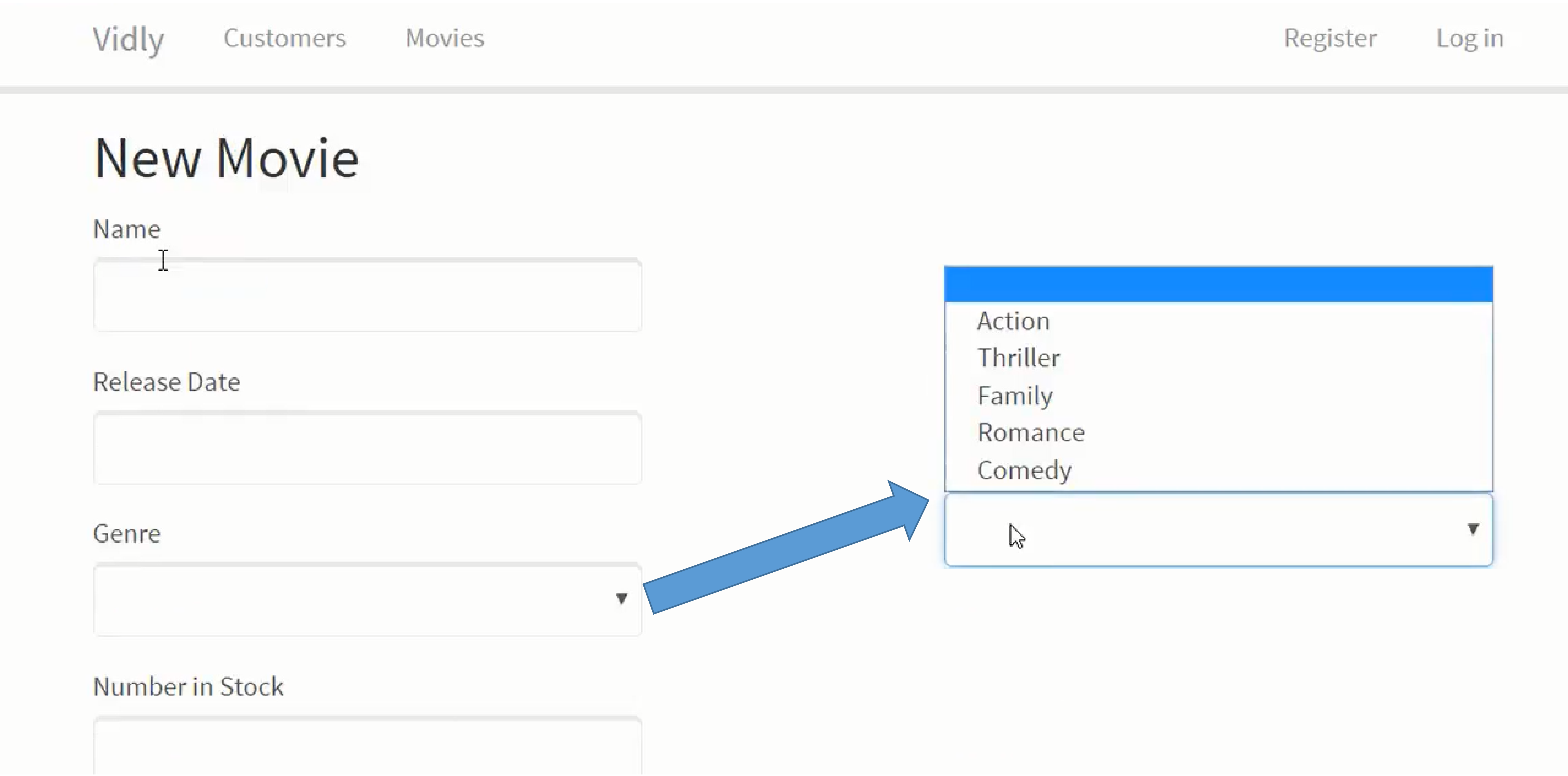
```
<div class="checkbox">
    <label>
        @Html.CheckBoxFor(m => m.Customer.IsSubscribedToNewsletter) Subscribed t
    </label>
</div>
@Html.HiddenFor(m => m.Customer.Id)
<button type="submit" class="btn btn-primary">Save</button>
```

Exercise



Movies

Movie	Genre
Hangover	Comedy
Die Hard	Action
The Terminator	Action
Toy Story	Family
Titanic	Romance
Zootopia	Family



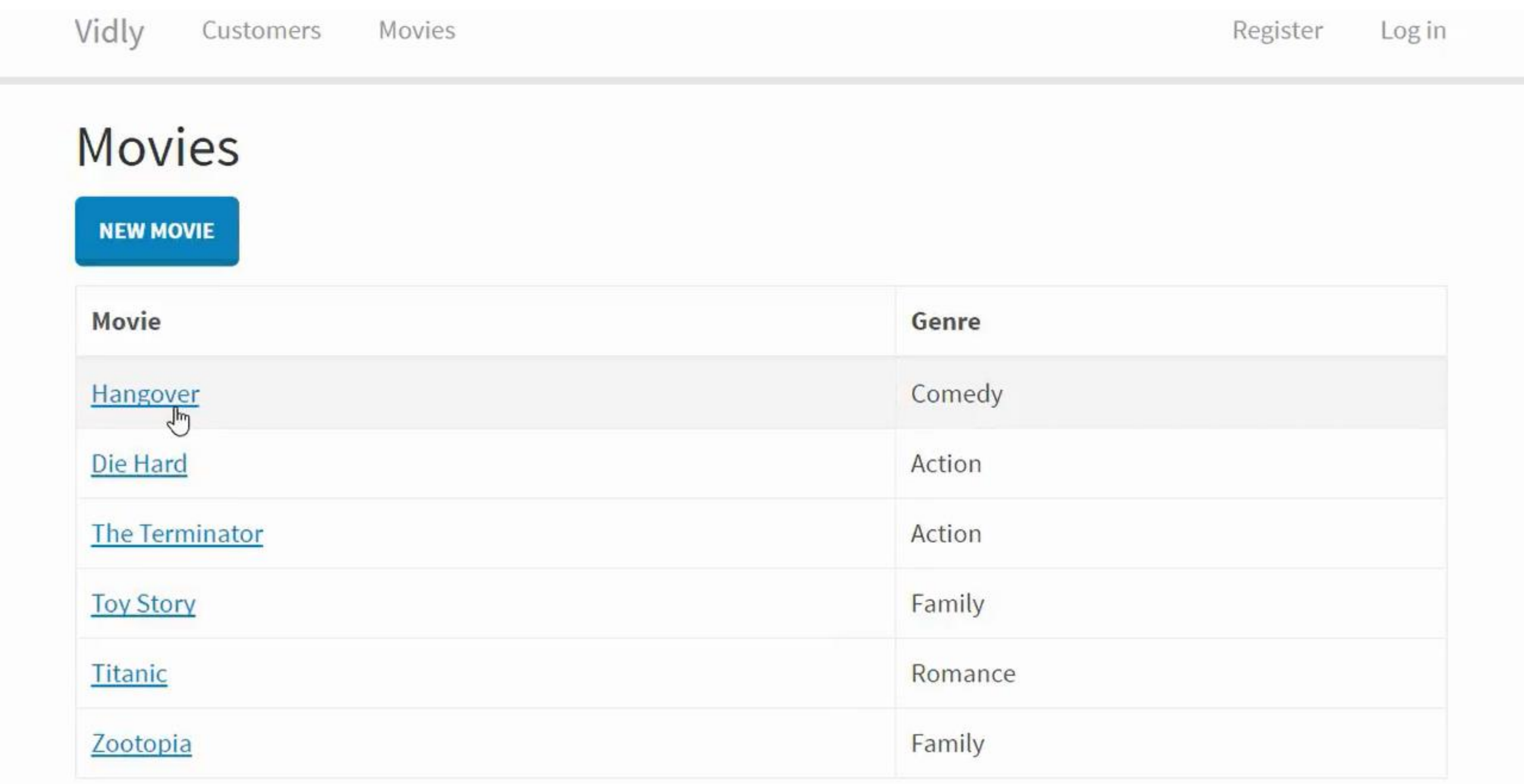
New Movie

Name

Release Date

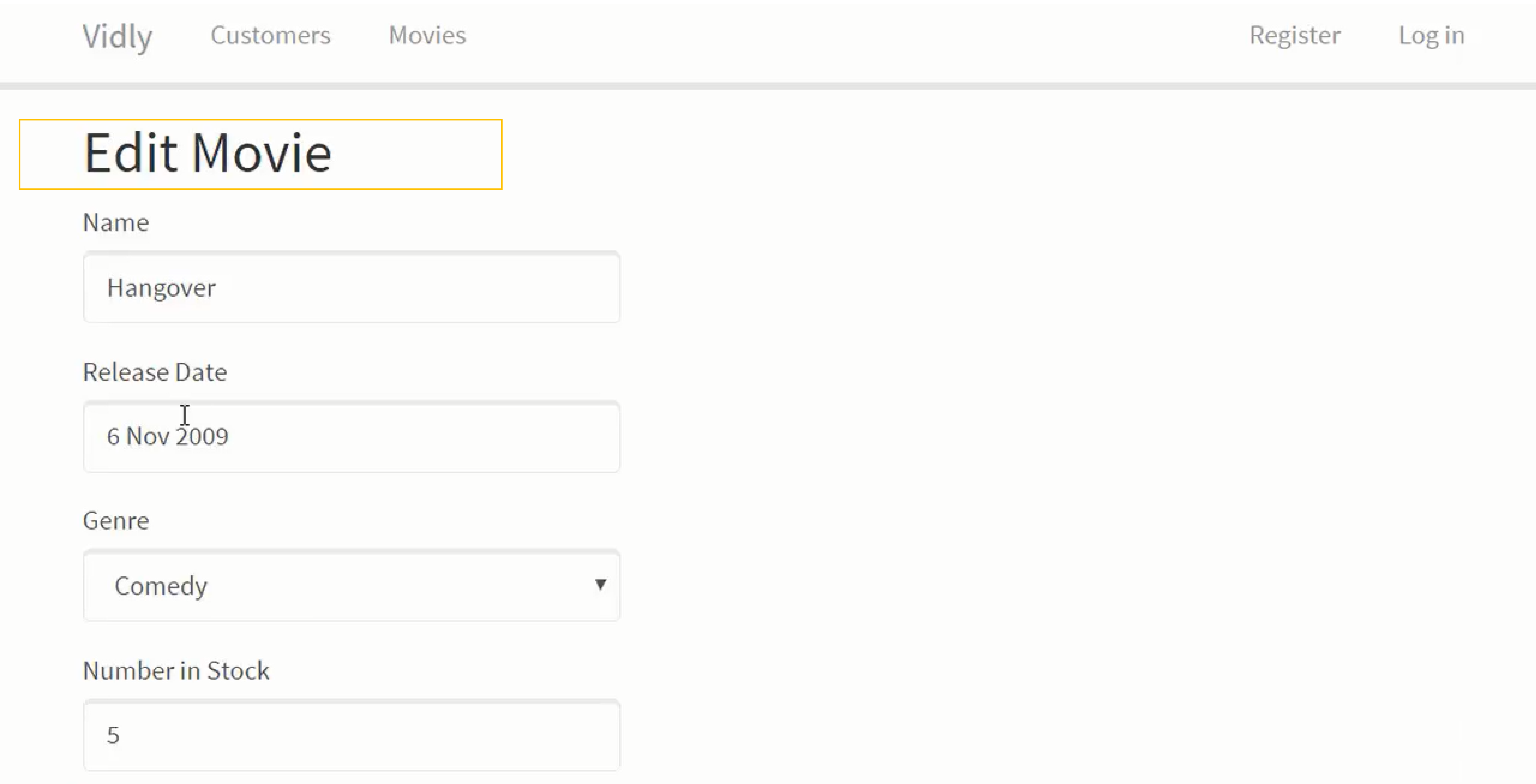
Genre

Number in Stock



Movies

Movie	Genre
Hangover	Comedy
Die Hard	Action
The Terminator	Action
Toy Story	Family
Titanic	Romance
Zootopia	Family



Edit Movie

Name

Hangover

Release Date

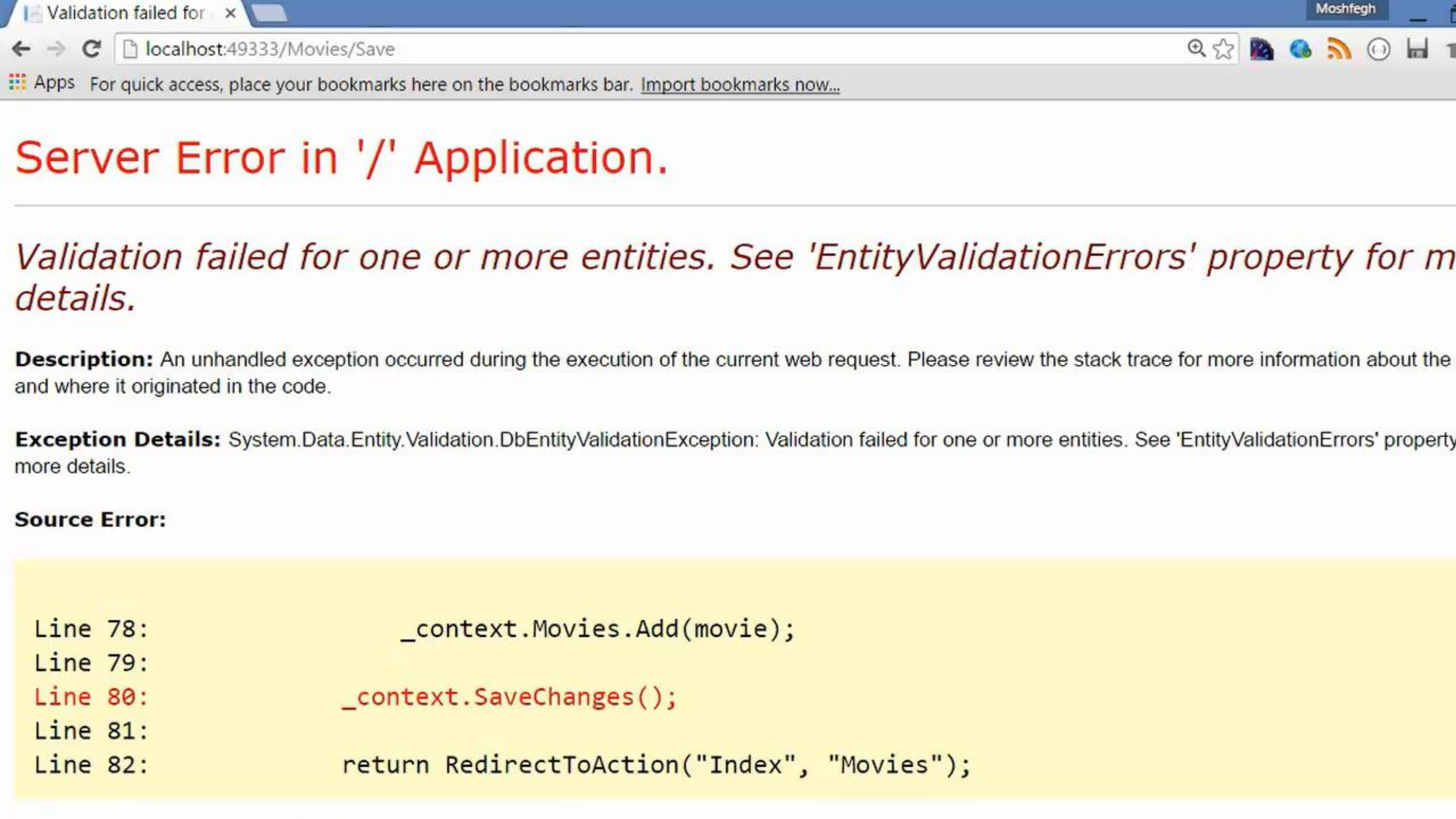
6 Nov 2009

Genre

Comedy ▾

Number in Stock

5


Validation failed for
Moshfegh
localhost:49333/Movies/Save
Apps
For quick access, place your bookmarks here on the bookmarks bar.
Import bookmarks now...

Server Error in '/' Application.

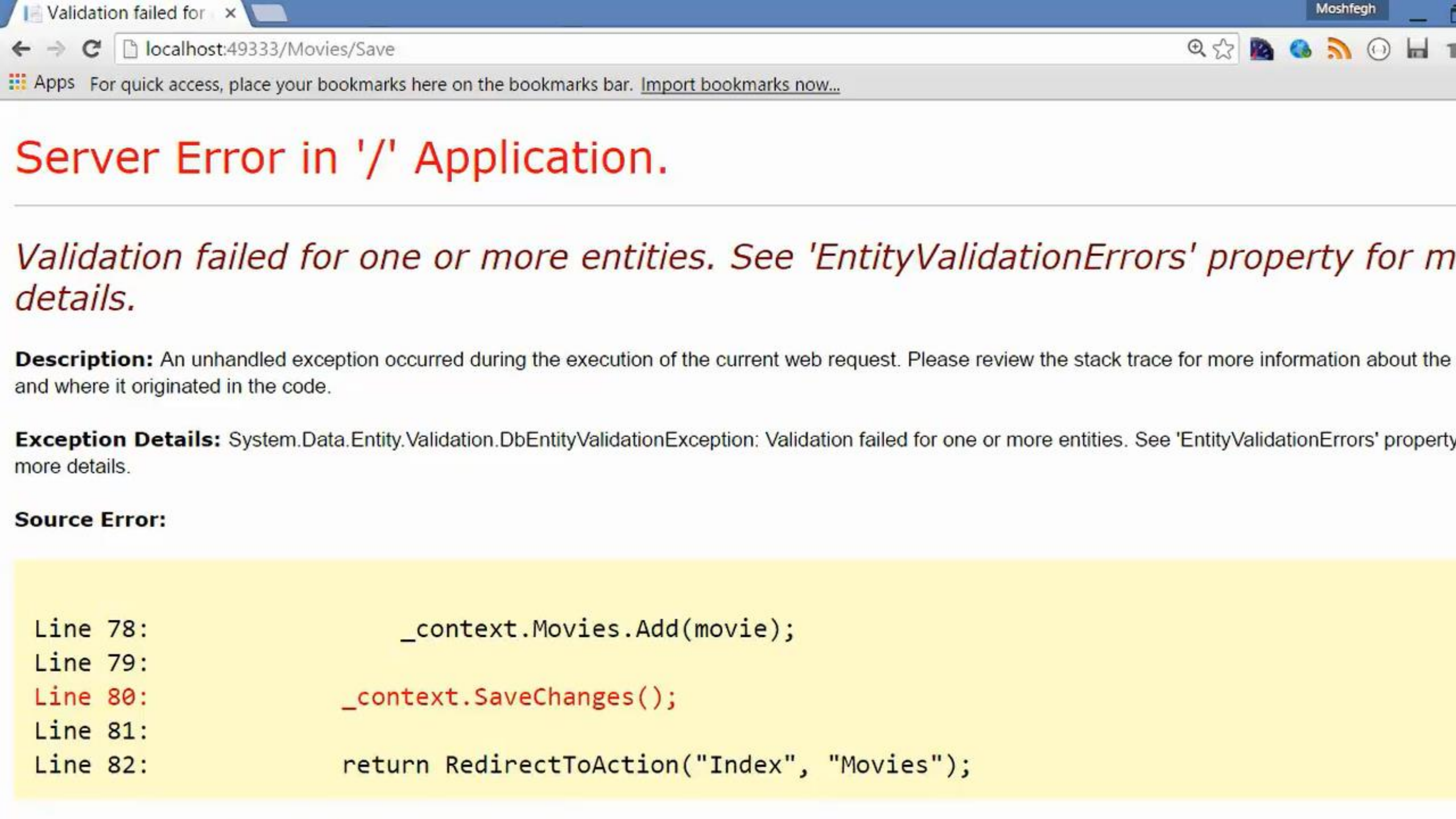
Validation failed for one or more entities. See 'EntityValidationErrors' property for m details.

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the and where it originated in the code.

Exception Details: System.Data.Entity.Validation.DbEntityValidationException: Validation failed for one or more entities. See 'EntityValidationErrors' property more details.

Source Error:

```
Line 78:                 _context.Movies.Add(movie);
Line 79:
Line 80:             _context.SaveChanges();
Line 81:
Line 82:             return RedirectToAction("Index", "Movies");
```


Validation failed for
Moshfegh
localhost:49333/Movies/Save
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

Server Error in '/' Application.

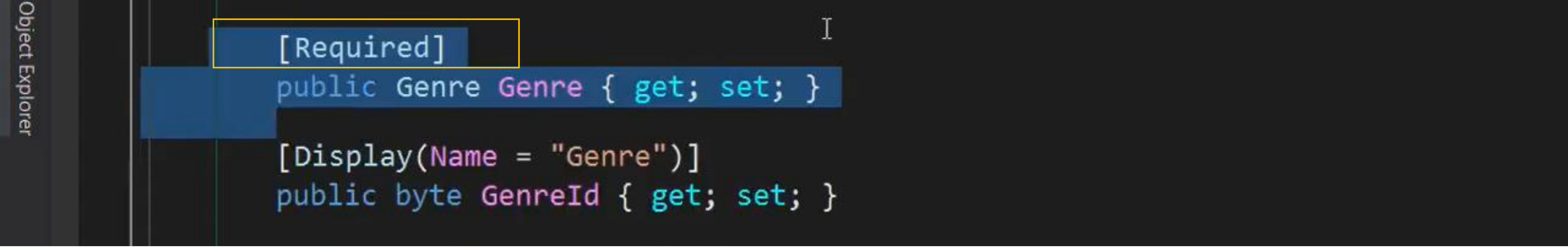
Validation failed for one or more entities. See 'EntityValidationErrors' property for m details.

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the and where it originated in the code.

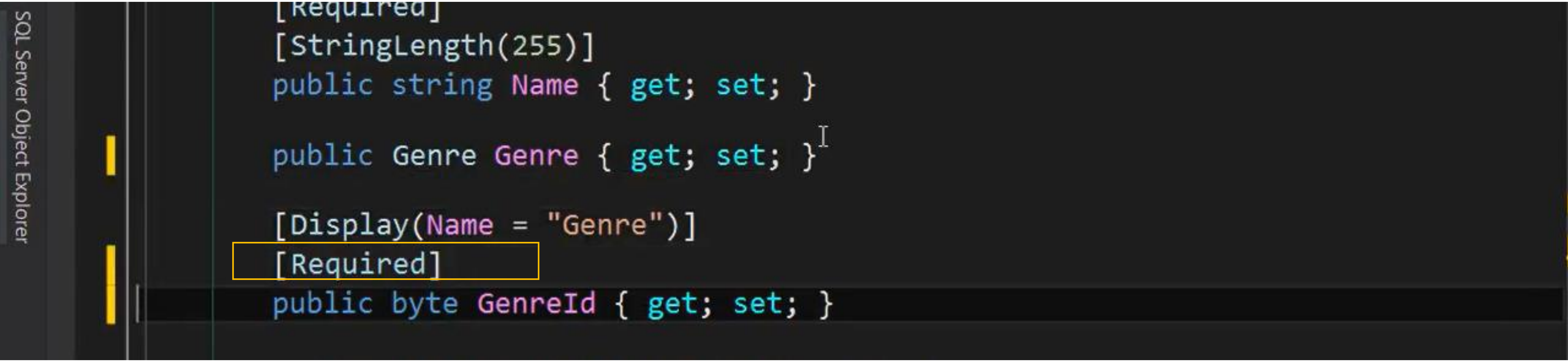
Exception Details: System.Data.Entity.Validation.DbEntityValidationException: Validation failed for one or more entities. See 'EntityValidationErrors' property more details.

Source Error:

```
Line 78:                 _context.Movies.Add(movie);
Line 79:
Line 80:             _context.SaveChanges();
Line 81:
Line 82:             return RedirectToAction("Index", "Movies");
```



```
[Required]
public Genre Genre { get; set; }

[Display(Name = "Genre")]
public byte GenreId { get; set; }
```



```
[Required]
[StringLength(255)]
public string Name { get; set; }

public Genre Genre { get; set; }

[Display(Name = "Genre")]
[Required]
public byte GenreId { get; set; }
```

Summary

Building Forms

Summary

- Markup
- Overriding labels
- Rendering drop-down lists
- Adding data
- Updating data

Next

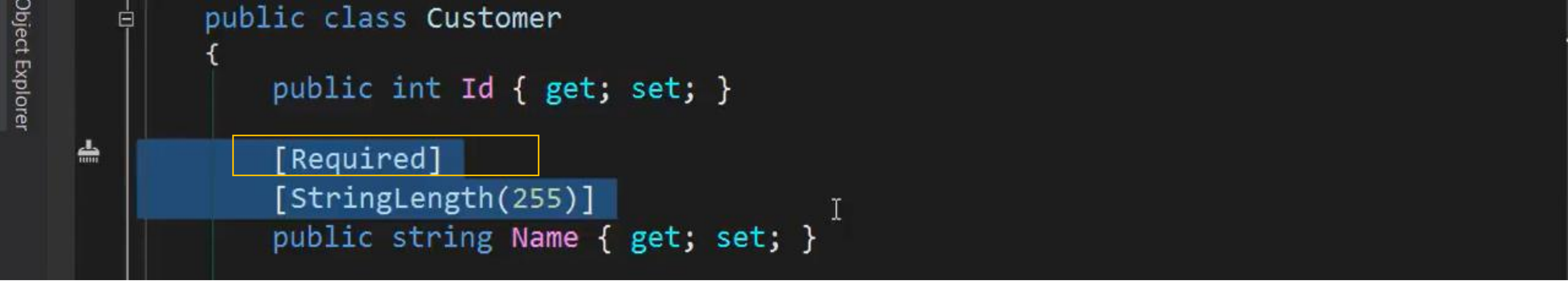
Validation

Implementing Validation

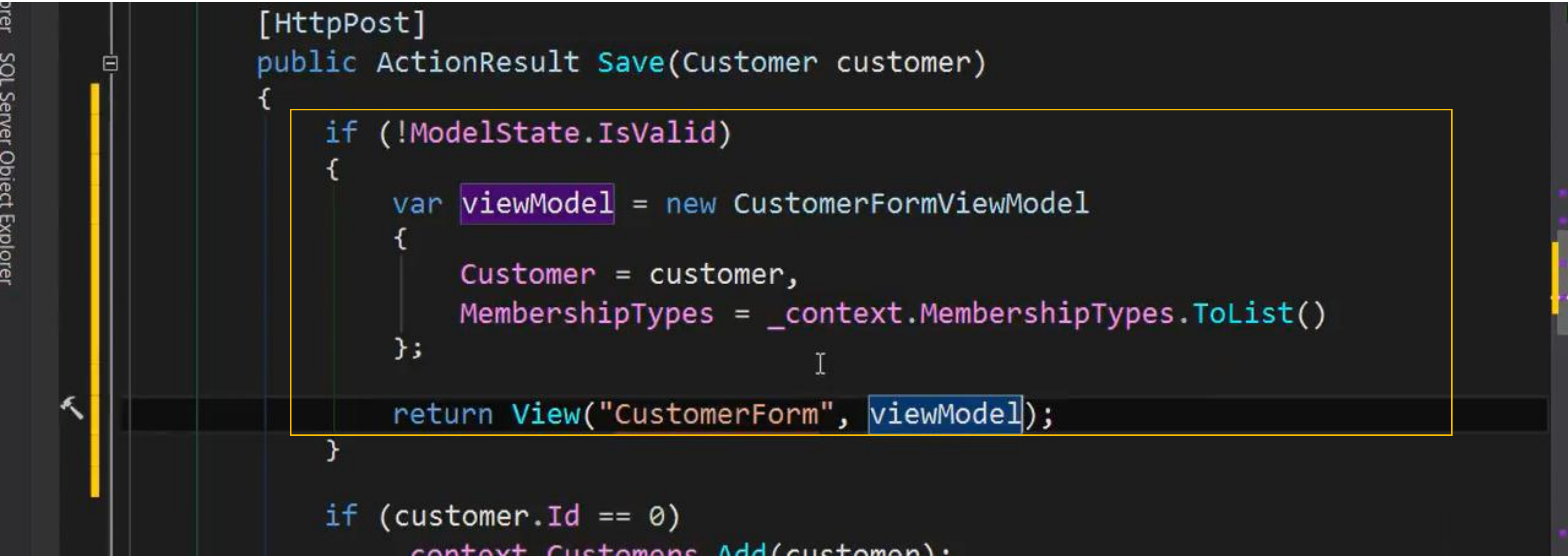
Adding Validation

Zouhair Rimale, Ph.D.

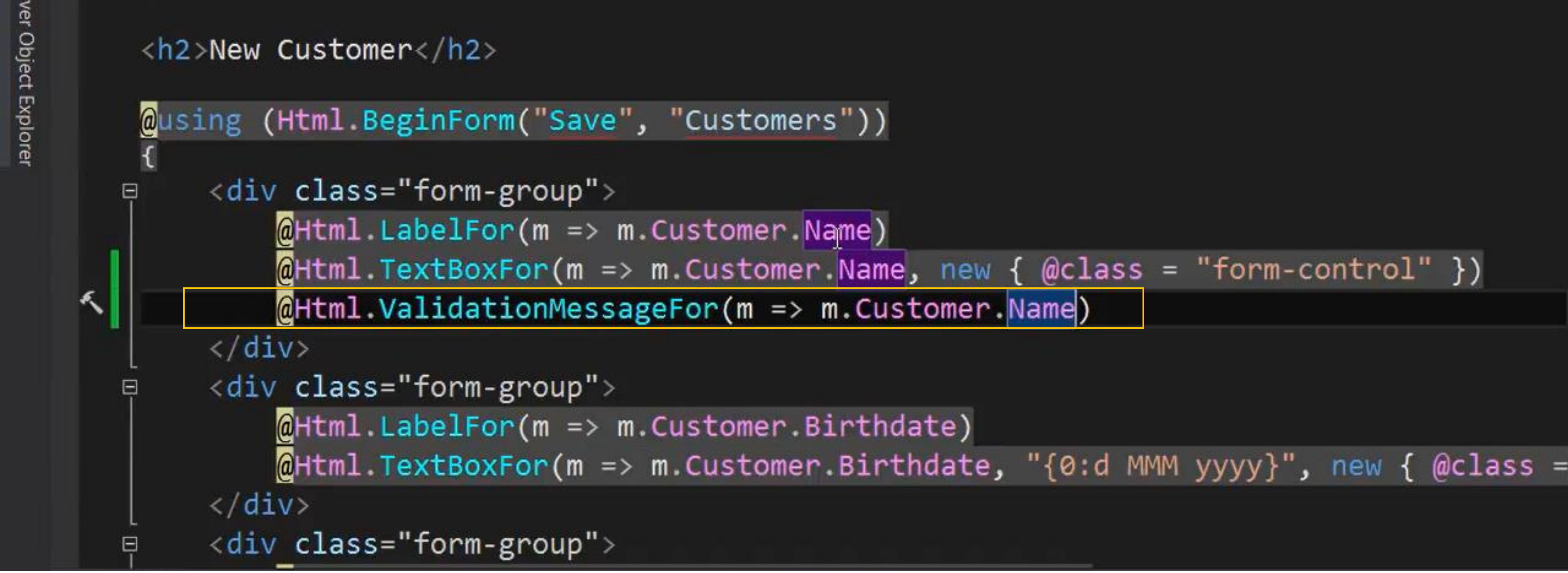
Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



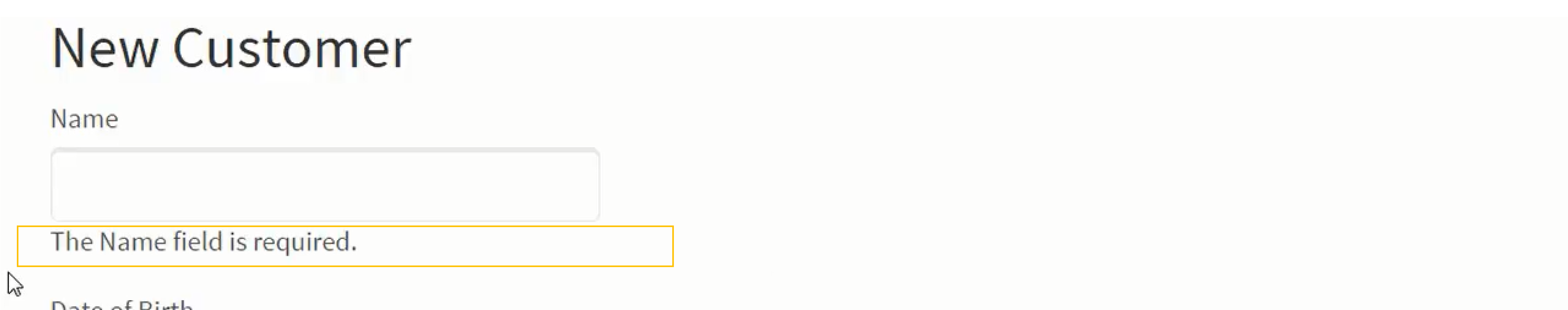
```
public class Customer
{
    public int Id { get; set; }

    [Required]
    [StringLength(255)]
    public string Name { get; set; }
```



```
[HttpPost]
public ActionResult Save(Customer customer)
{
    if (!ModelState.IsValid)
    {
        var viewModel = new CustomerFormViewModel
        {
            Customer = customer,
            MembershipTypes = _context.MembershipTypes.ToList()
        };

        return View("CustomerForm", viewModel);
    }

    if (customer.Id == 0)
        _context.Customers.Add(customer);
```



```
<h2>New Customer</h2>

@using (Html.BeginForm("Save", "Customers"))
{
    <div class="form-group">
        @Html.LabelFor(m => m.Customer.Name)
        @Html.TextBoxFor(m => m.Customer.Name, new { @class = "form-control" })
        @Html.ValidationMessageFor(m => m.Customer.Name)
    </div>
    <div class="form-group">
        @Html.LabelFor(m => m.Customer.Birthdate)
        @Html.TextBoxFor(m => m.Customer.Birthdate, "{0:d MMM yyyy}", new { @class =
    </div>
    <div class="form-group">
```



New Customer

Name

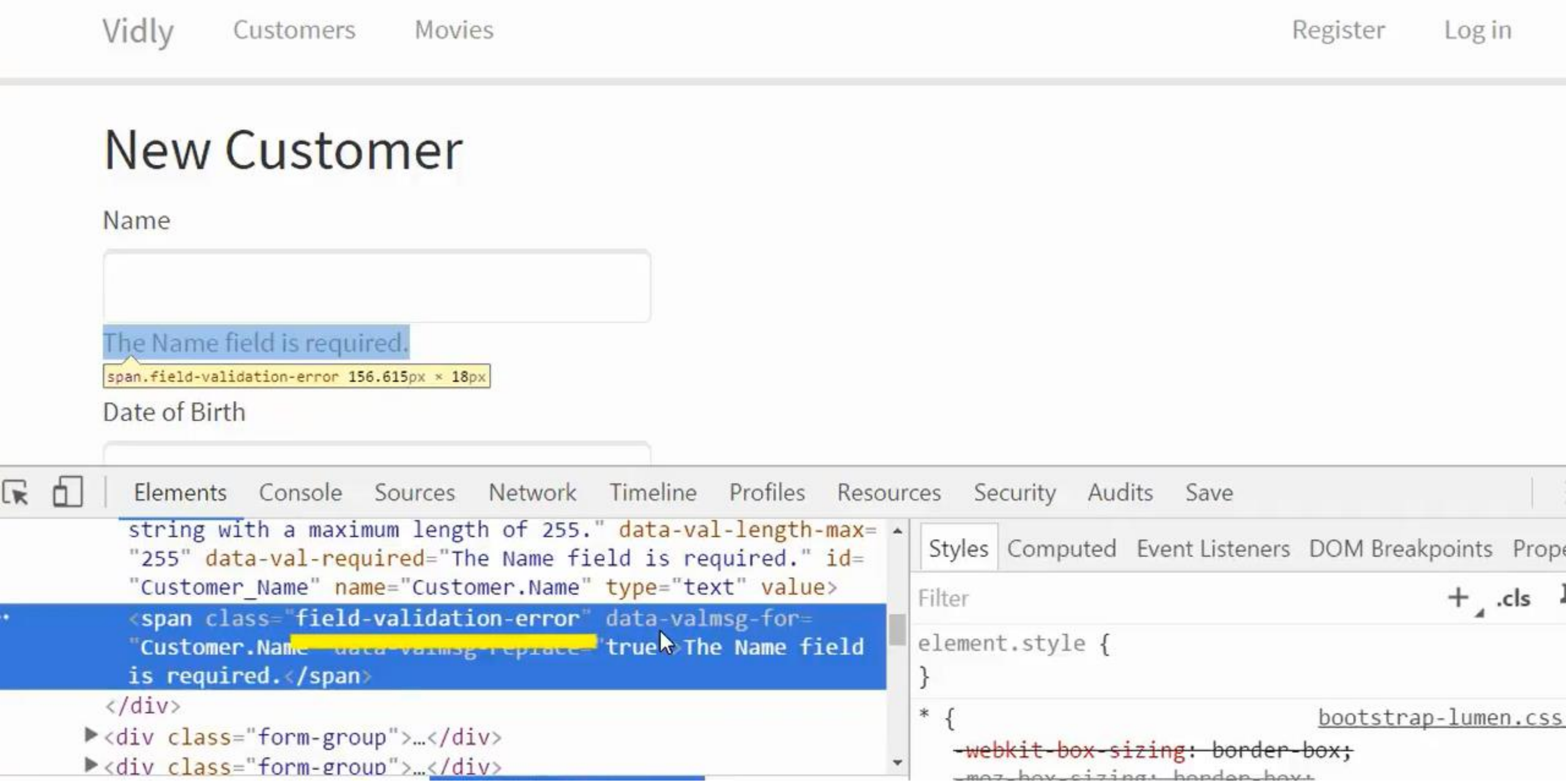
The Name field is required.

Date of Birth

Styling Validation Errors

Zouhair Rimale, Ph.D.

Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular



New Customer

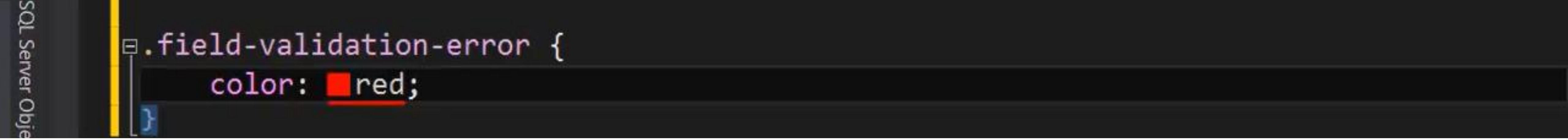
Name

The Name field is required.

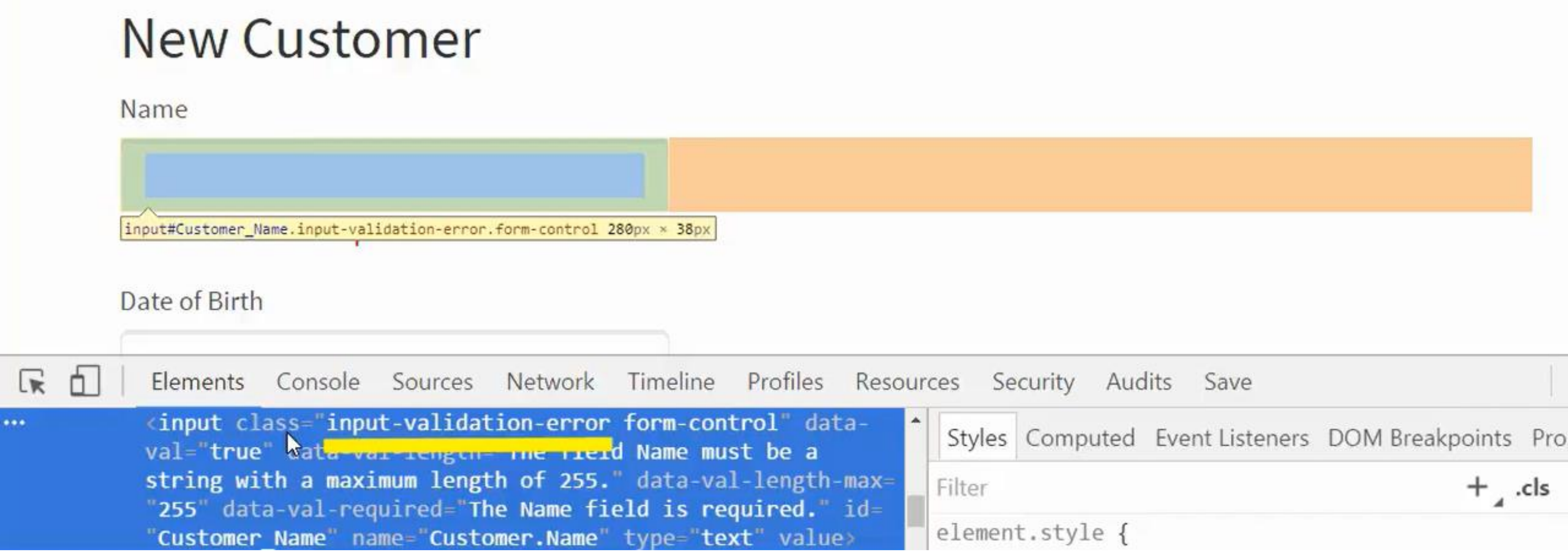
Date of Birth

```
string with a maximum length of 255." data-val-length-max=
"255" data-val-required="The Name field is required." id=
"Customer_Name" name="Customer.Name" type="text" value>
<span class="field-validation-error" data-valmsg-for=
"Customer.Name" data-valmsg-replace="true">The Name field
is required.</span>
</div>
▶<div class="form-group">…</div>
▶<div class="form-group">…</div>
```

```
element.style {
}
* {                                   bootstrap-lumen.css
  ~~-webkit-box-sizing: border-box;~~
  ~~-moz-box-sizing: border-box;~~
```



```
.field-validation-error {
    color: red;
}
```



New Customer

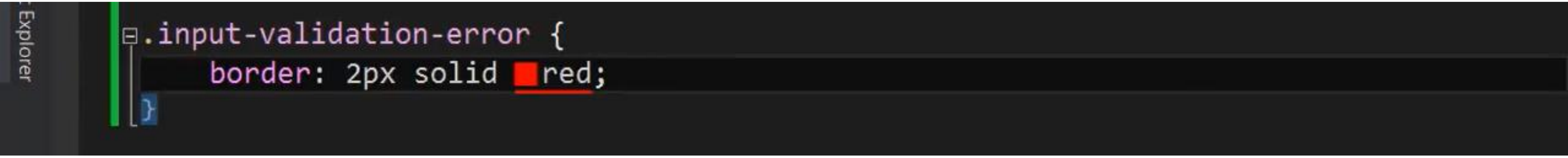
Name

input#Customer_Name.input-validation-error.form-control 280px × 38px

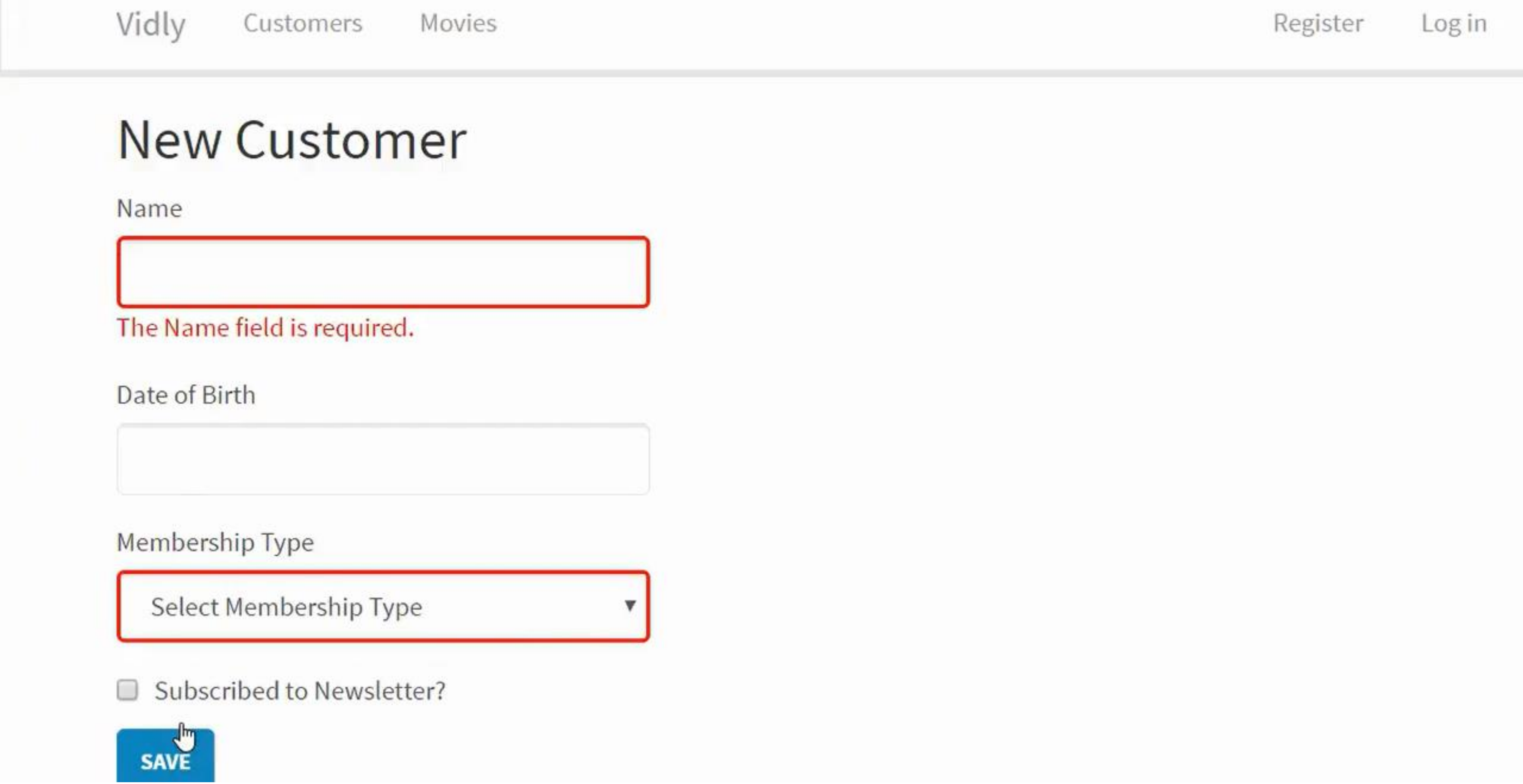
Date of Birth

```
<input class="input-validation-error form-control" data-val="true" data-val-length="The field Name must be a string with a maximum length of 255." data-val-length-max="255" data-val-required="The Name field is required." id="Customer_Name" name="Customer.Name" type="text" value>
```

```
element.style {
```



```
.input-validation-error {
    border: 2px solid red;
}
```



New Customer

Name

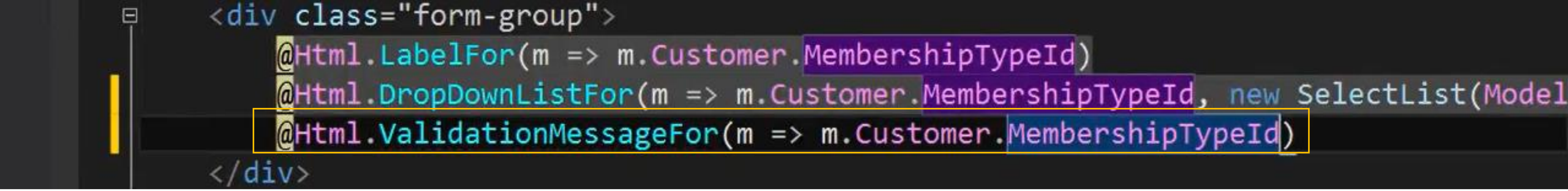
The Name field is required.

Date of Birth

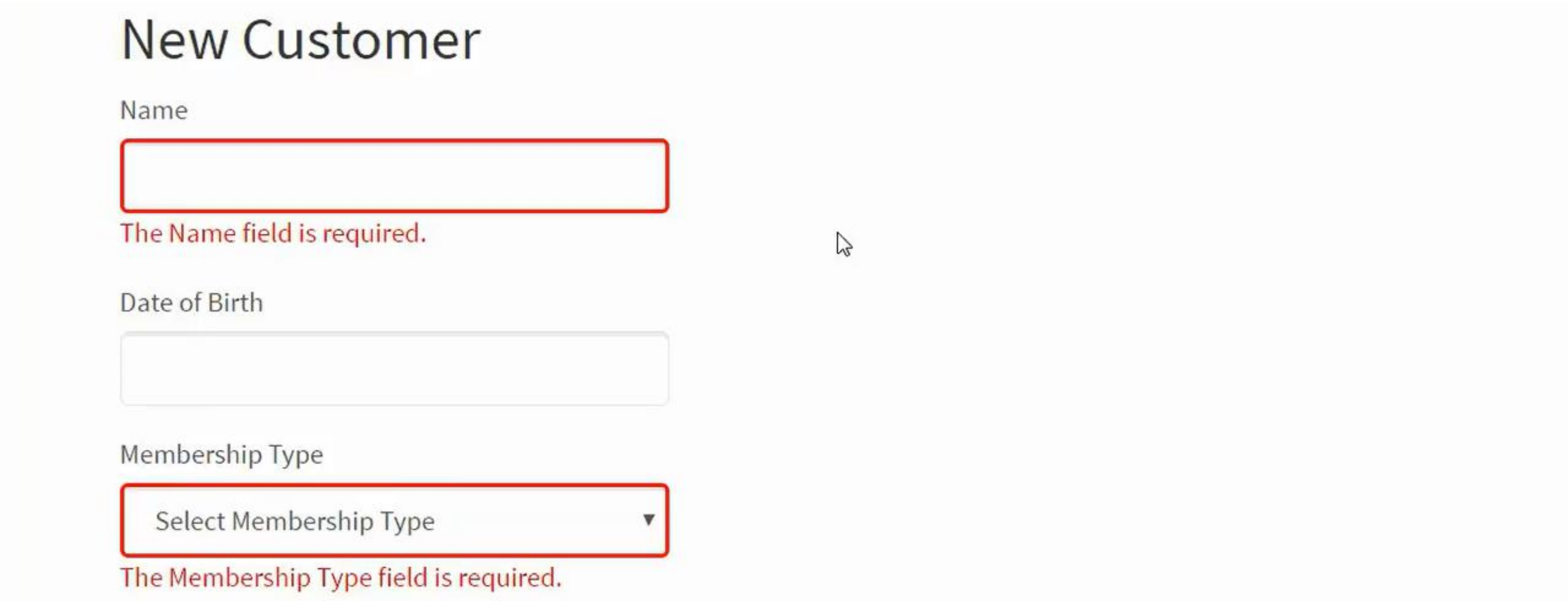
Membership Type

Select Membership Type ▾

□ Subscribed to Newsletter?



```
<div class="form-group">
    @Html.LabelFor(m => m.Customer.MembershipTypeId)
    @Html.DropDownListFor(m => m.Customer.MembershipTypeId, new SelectList(Model
    @Html.ValidationMessageFor(m => m.Customer.MembershipTypeId)
</div>
```



New Customer

Name

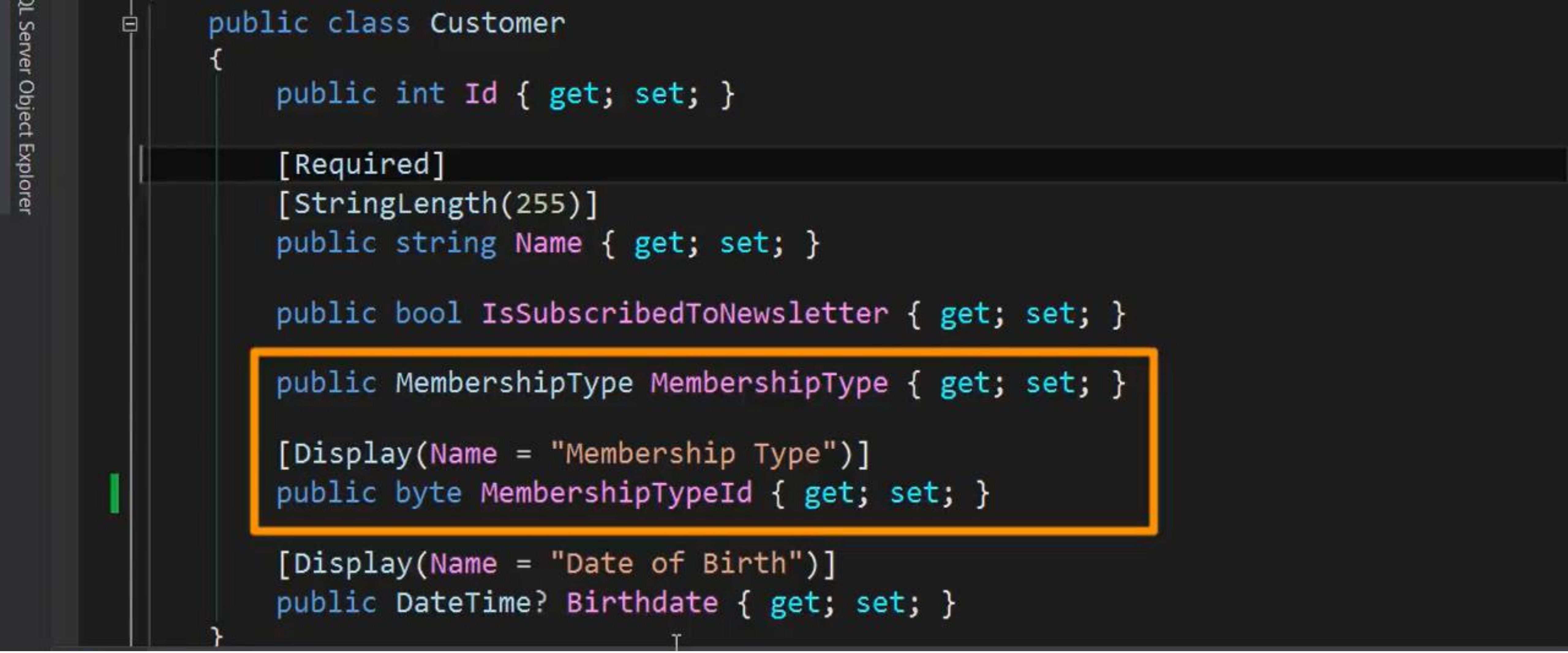
The Name field is required.

Date of Birth

Membership Type

Select Membership Type ▾

The Membership Type field is required.



```
public class Customer
{
    public int Id { get; set; }

    [Required]
    [StringLength(255)]
    public string Name { get; set; }

    public bool IsSubscribedToNewsletter { get; set; }

    public MembershipType MembershipType { get; set; }

    [Display(Name = "Membership Type")]
    public byte MembershipTypeId { get; set; }

    [Display(Name = "Date of Birth")]
    public DateTime? Birthdate { get; set; }
}
```

Data Annotations

Zouhair Rimale, Ph.D.

Expert technique .NET | SharePoint | ASP.NET MVC | WEB API | Angular