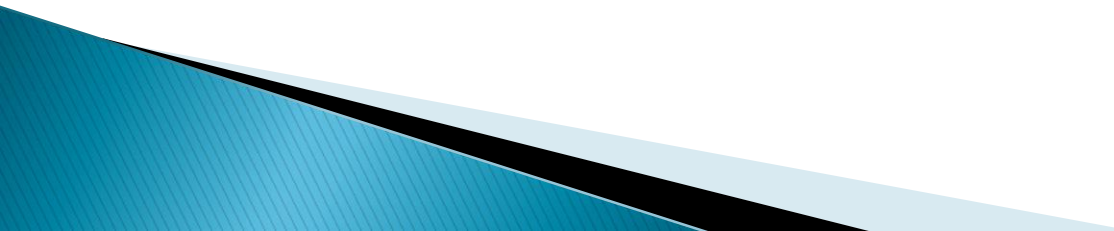# Enterprise Frameworks

Authentication Tutorial

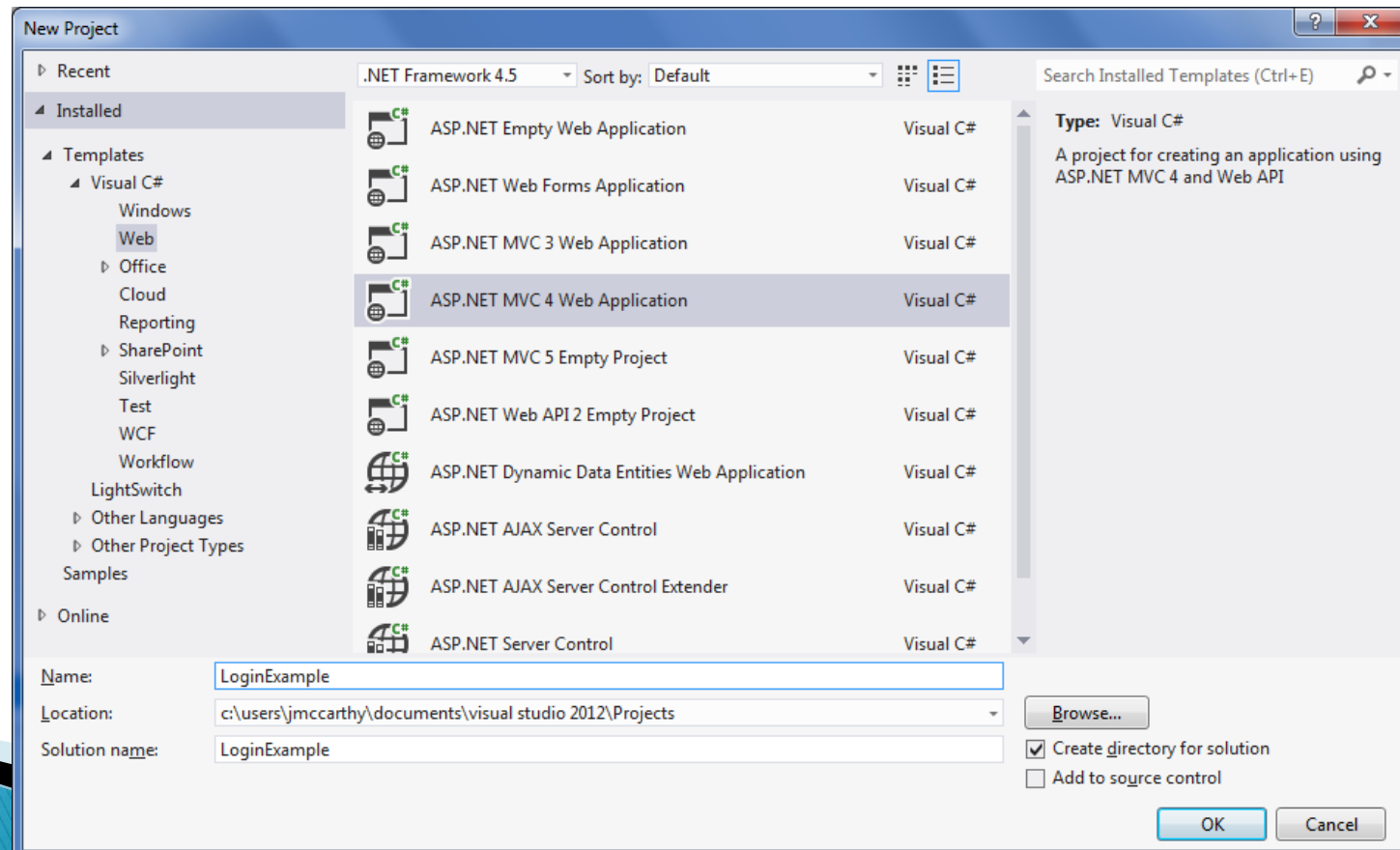# Overview

- Add a login to a project
- General User
- Admin User
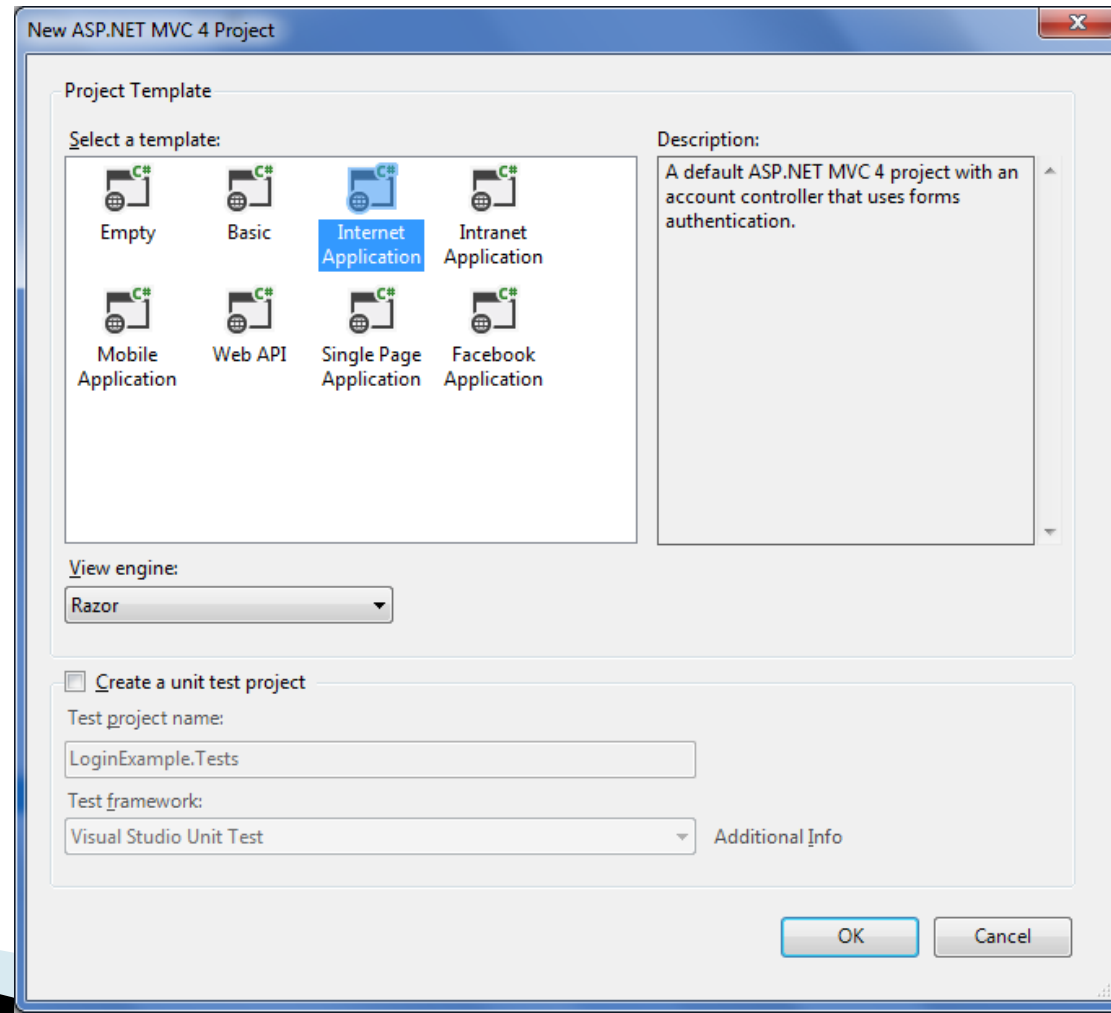
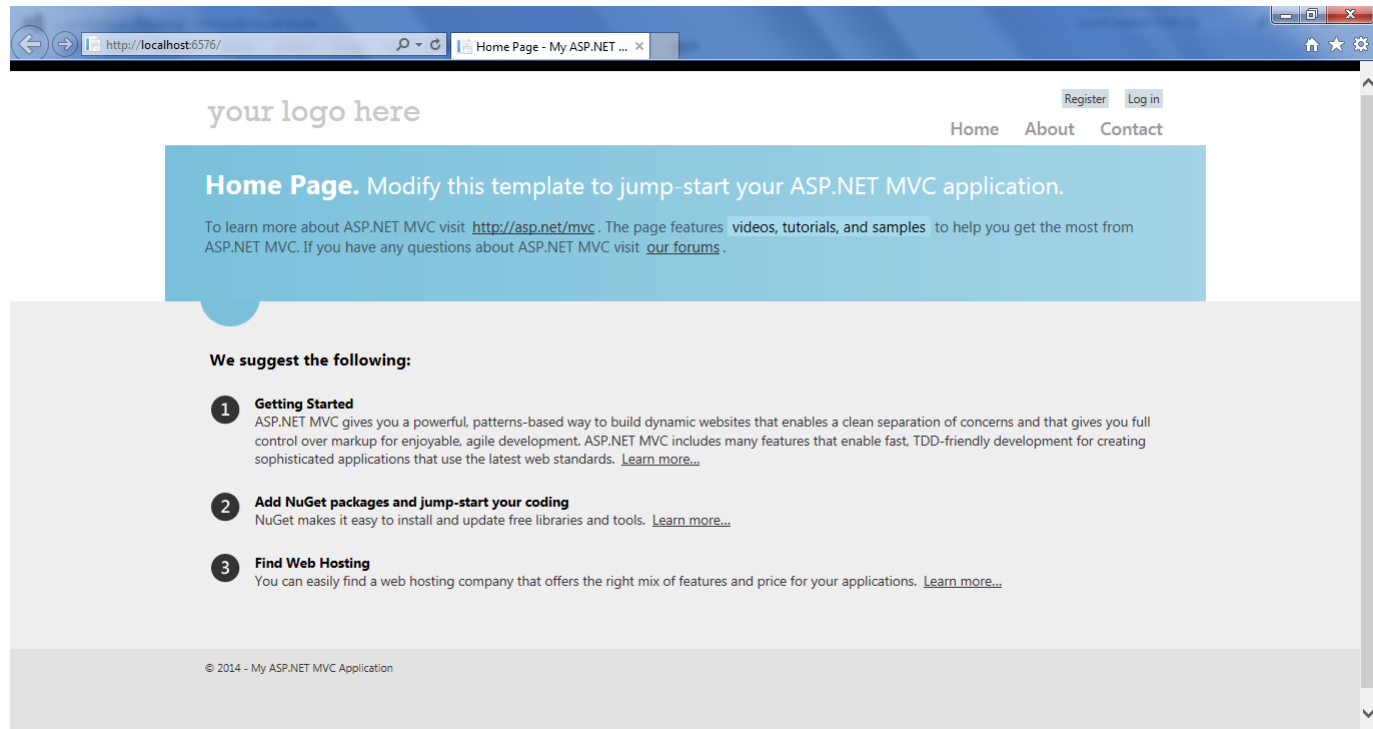# Create a new web application

- ASP.Net MVC 4 Application
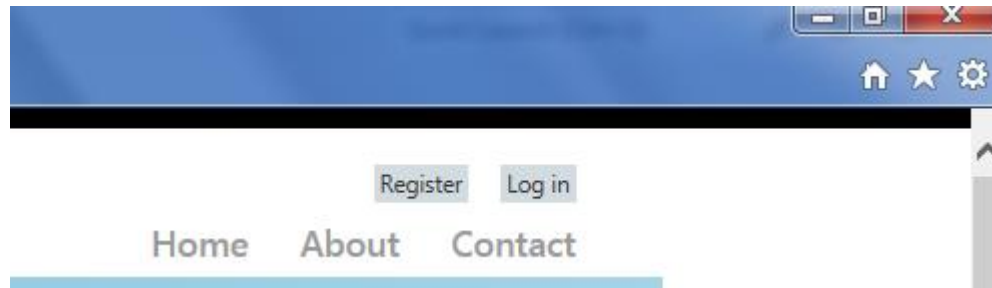
Create a new web application

▸ Internet Application

Run the Application
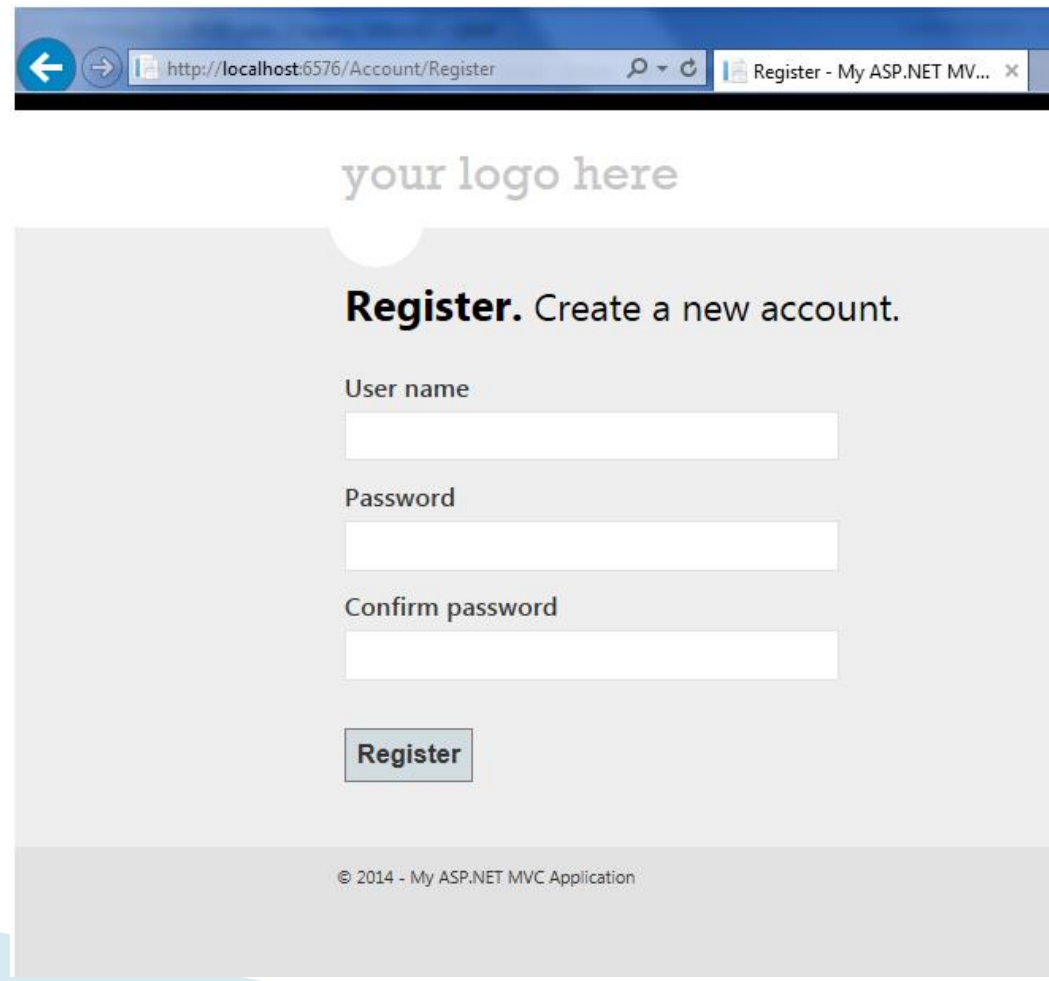
▸ The running application should look as follows

Register a new User

- Register a new user

Register a new User

▸ The registration form looks as follows:

Register a new User

▸ Enter a username and a password

Register a new User

- A successful registration will result is a screen that looks as follows:
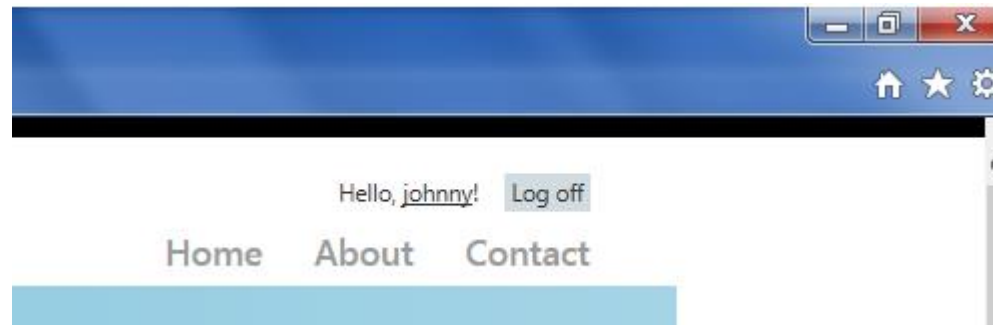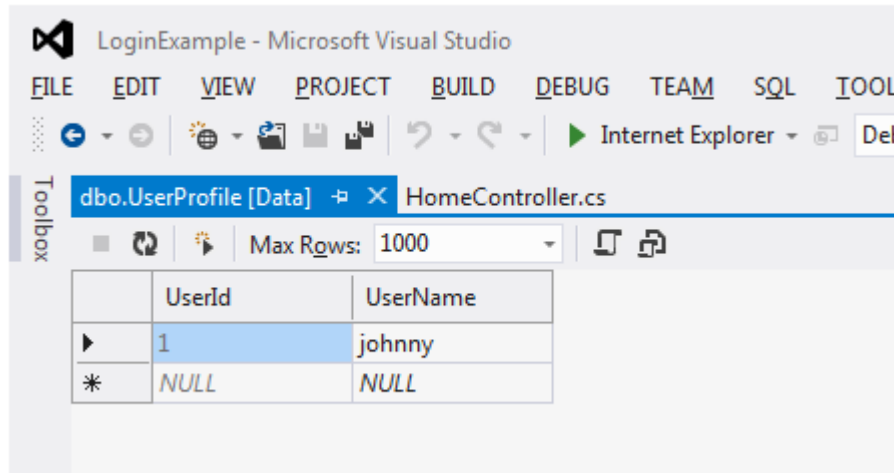
View the Database

- Creating the new user has setup a database to cater for the authentication process.
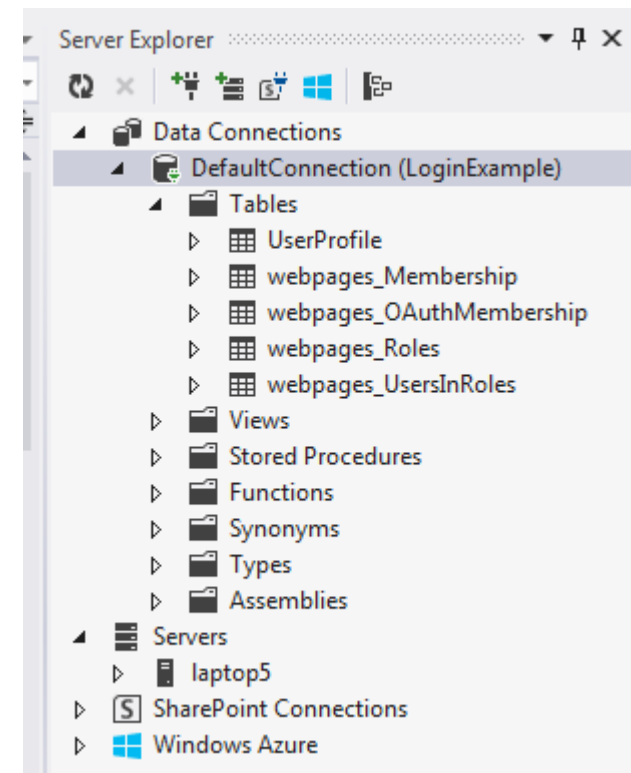
View Data

- Right click UserProfile and select show data.
- The following will appear:

View Data
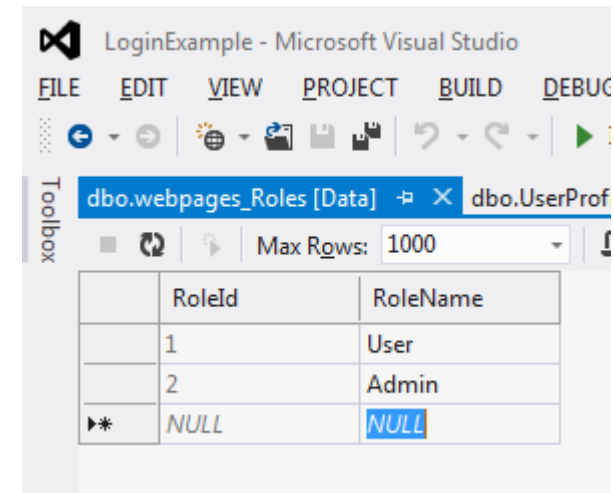
- Right click webpages_Roles and select show data.
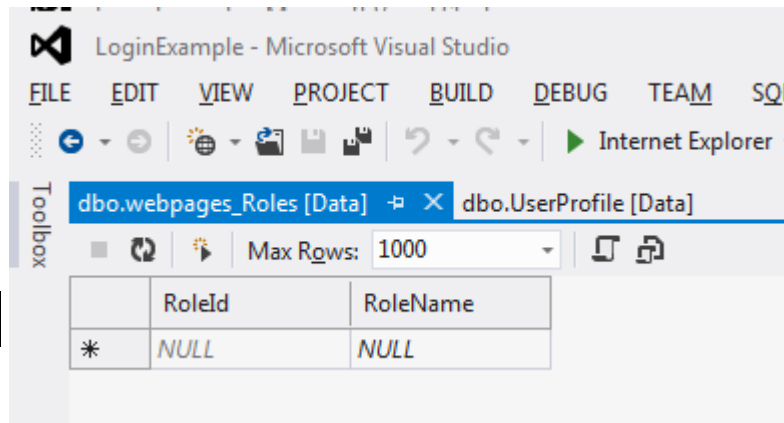
- We wil

Add new Roles

▸ Add two new Roles:

▸ User
▸ Admin

Associate a user with a role

- Right click webpages_UsersInRoles and select show data.
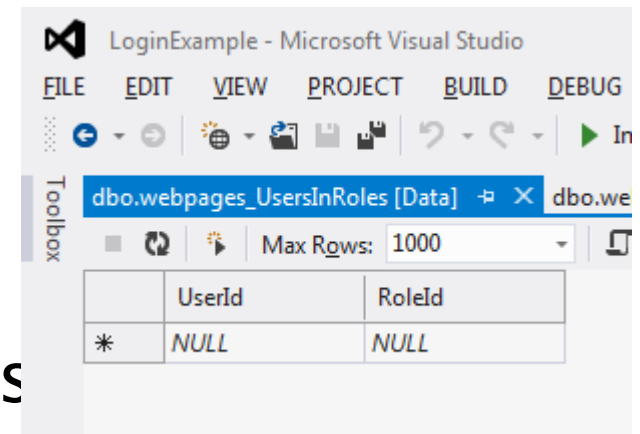
- Add an entry to associate a us
- Setting user Johnny as an Admin
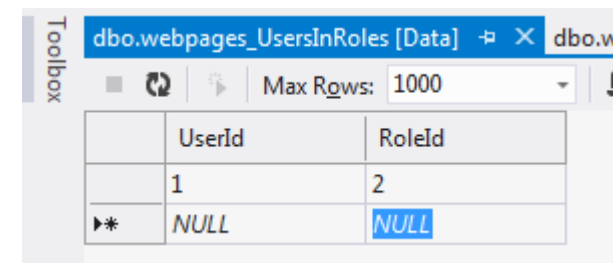
Testing the new user settings

▸ Register a new user through the web application named billy.

▸ The application should now have two users:
  ◦ A standard user named billy
  ◦ An Admin user named Johnny

Protecting Functionality

- For the standard web application the about page can only be viewed be an Admin User, the Contact page can be viewed by a general user and an Admin, and the home page can be viewed even if a user hasn't logged in.

Adding a filter

- Add a filter to the FilterConfig file in the App_Start folder. This filter will require a user to login to use the functionality of the web application.
- Add: filters.Add(new AuthorizeAttribute());
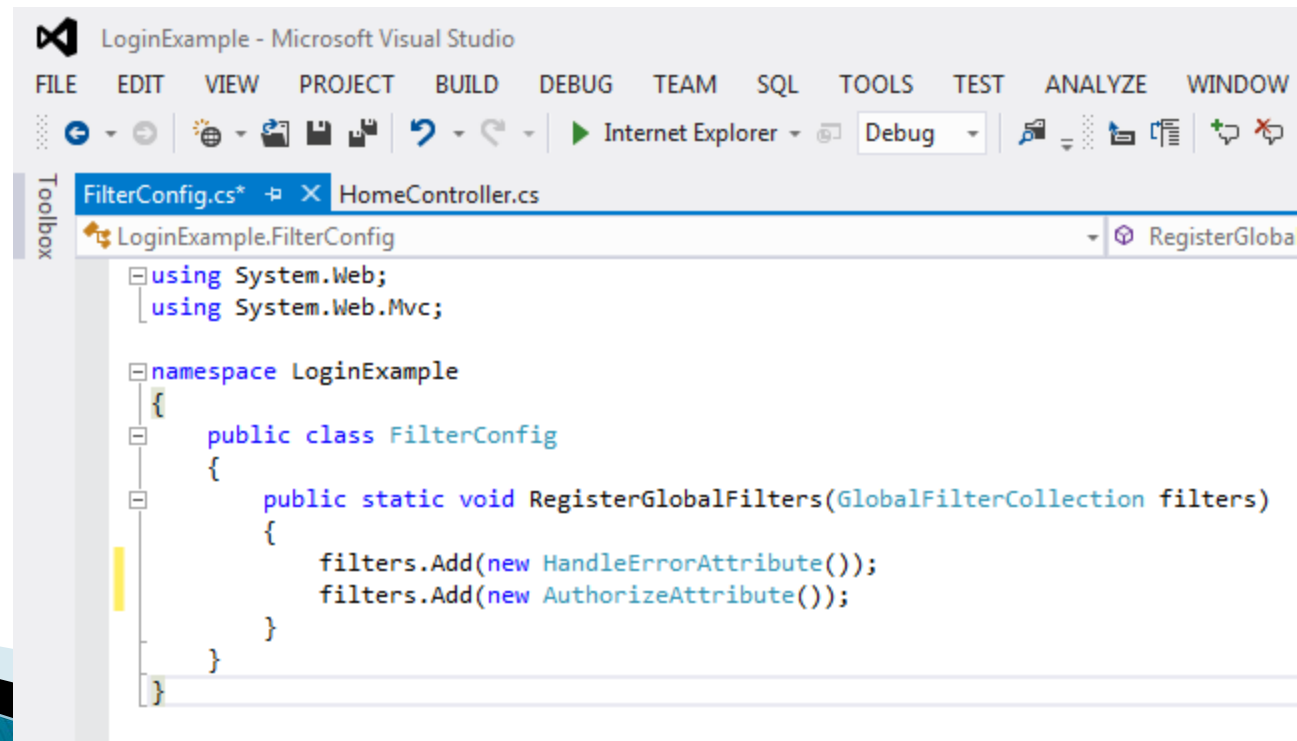
Update the Home Controller

- Step 1 – Allow anyone view the home page:
- Add [AllowAnonymous] to the top of the Index method.

```
[AllowAnonymous]
public ActionResult Index()
{
    ViewBag.Message = "Modify this template to jump-start your ASP.NET MVC application.";

    return View();
}
```

Update the Home Controller

▸ Step 2 – Allow only Admins view the About page. Add [Authorize(Roles="Admin")] to the top of the About method.

```
[Authorize(Roles="Admin")]
public ActionResult About()
{
    ViewBag.Message = "Your app description page.";

    return View();
}
```

Update the Home Controller

▸ Step 3 – Allow logged in users view the Contact page. This is a requirement by the filter that was added earlier, no action is needed to provide the functionality in the web app

```
// Protected by the default filter
public ActionResult Contact()
{
    ViewBag.Message = "Your contact page.";

    return View();
}
```

Run the application and test the functionality.

- Additional tasks.
- Try add this functionality to the Associations Lab with books and publishers.