Create Database first

How to do that

CREATE TABLE [dbo].[System\_Users]

(

[Id] INT NOT NULL IDENTITY ,

[Username] NVARCHAR(50) NOT NULL,

[Password] NVARCHAR(MAX) NOT NULL,

[RegDate] DATETIME NOT NULL DEFAULT CURRENT\_TIMESTAMP,

[Email] NVARCHAR(50) NOT NULL,

PRIMARY KEY ([Id])

)

GO

CREATE INDEX [IX\_System\_Users\_Username] ON [dbo].[System\_Users] ([Username])

GO

INSERT INTO [dbo].[System\_Users]

([Username], [Password], [Email])

VALUES

('test', 'a94a8fe5ccb19ba61c4c0873d391e987982fbbd3', 'test@test.test')

GO

1. Controller Class

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using System.Web.Security;

namespace Creating\_a\_custom\_user\_login\_form.Controllers

{

public class UserController : Controller

{

*//*

*// GET: /User/*

public ActionResult Index()

{

return View();

}

[HttpGet]

public ActionResult Login()

{

return View();

}

[HttpPost]

public ActionResult Login(Models.User user)

{

if (ModelState.IsValid)

{

if (user.IsValid(user.UserName, user.Password))

{

FormsAuthentication.SetAuthCookie(user.UserName, user.RememberMe);

return RedirectToAction("Index", "Home");

}

else

{

ModelState.AddModelError("", "Login data is incorrect!");

}

}

return View(user);

}

public ActionResult Logout()

{

FormsAuthentication.SignOut();

return RedirectToAction("Index", "Home");

}

}

}

1. Create Model and add this class

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Threading.Tasks;

namespace Creating\_a\_custom\_user\_login\_form.Models

{

public class User

{

[Required]

[Display(Name = "User name")]

public string UserName { get; set; }

[Required]

[DataType(DataType.Password)]

[Display(Name = "Password")]

public string Password { get; set; }

[Display(Name = "Remember on this computer")]

public bool RememberMe { get; set; }

/// *<summary>*

/// *Checks if user with given password exists in the database*

/// *</summary>*

/// *<param name="\_username">User name</param>*

/// *<param name="\_password">User password</param>*

/// *<returns>True if user exist and password is correct</returns>*

public bool IsValid(string \_username, string \_password)

{

using (var cn = new SqlConnection(@"Data Source=(LocalDB)\v11.0;AttachDbFilename" +

@"='C:\Tutorials\1 - Creating a custom user login form\Creating " +

@"a custom user login form\App\_Data\Database1.mdf';Integrated Security=True"))

{

string \_sql = @"SELECT [Username] FROM [dbo].[System\_Users] " +

@"WHERE [Username] = @u AND [Password] = @p";

var cmd = new SqlCommand(\_sql, cn);

cmd.Parameters

.Add(new SqlParameter("@u", SqlDbType.NVarChar))

.Value = \_username;

cmd.Parameters

.Add(new SqlParameter("@p", SqlDbType.NVarChar))

.Value = Encode(\_password);

cn.Open();

var reader = cmd.ExecuteReader();

if (reader.HasRows)

{

reader.Dispose();

cmd.Dispose();

return true;

}

else

{

reader.Dispose();

cmd.Dispose();

return false;

}

}

}

public static string Encode(string value)

{

var hash = System.Security.Cryptography.SHA1.Create();

var encoder = new System.Text.ASCIIEncoding();

var combined = encoder.GetBytes(value ?? "");

return BitConverter.ToString(hash.ComputeHash(combined)).ToLower().Replace("-", "");

}

}

}

4)

Create View

Login.cshtml

@model Creating\_a\_custom\_user\_login\_form.Models.User

@{

ViewBag.Title = "Login";

Layout = "~/Views/Shared/\_Layout.cshtml";

}

@using (Html.BeginForm())

{

@Html.ValidationSummary(true, "Login failed. Check your login details.");

<div>

<fieldset>

<legend>Login</legend>

<div class="editor-label">

@Html.LabelFor(u => u.UserName)

</div>

<div class="editor-field">

@Html.TextBoxFor(u => u.UserName)

@Html.ValidationMessageFor(u => u.UserName)

</div>

<div class="editor-label">

@Html.LabelFor(u => u.Password)

</div>

<div class="editor-field">

@Html.PasswordFor(u => u.Password)

@Html.ValidationMessageFor(u => u.Password)

</div>

<div class="editor-label">

@Html.CheckBoxFor(u => u.RememberMe)

@Html.LabelFor(u => u.RememberMe)

</div>

<input type="submit" value="Log In" />

</fieldset>

</div>

}

\_Layout.cshtml

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width" />

<title>@ViewBag.Title</title>

@Styles.Render("~/Content/css")

@Scripts.Render("~/bundles/modernizr")

</head>

<body>

<div style="width: auto; background-color: #728ea7;">

@if (Request.IsAuthenticated) {

<strong>@Html.Encode(User.Identity.Name)</strong>

@Html.ActionLink("Sign Out", "Logout", "User")

}

else {

@Html.ActionLink("Register", "Register", "User")

<span> | </span>

@Html.ActionLink("Sign In", "Login", "User")

}

</div>

@RenderBody()

@Scripts.Render("~/bundles/jquery")

@RenderSection("scripts", required: false)

</body>

</html>