

Question 2

1. What are HTML helpers in MVC?

Html is a property of the HtmlHelper class included in base class of razor viewWebViewPage. The ActionLink() and DisplayNameFor() are extension methods included in the HtmlHelper class.

The HtmlHelper class generates HTML elements.

For example, @Html.ActionLink("Create New", "Create") would generate anchor tag Create New.

2. What is Razor in MVC?

Razor view engine in ASP.NET MVC is the syntax that allows you to write server-side code on view. It helps to combine code and HTML in a fluid manner. Razor is not a new programming language. If you know C#, Vb.Net, and bit HTML, you can easily write Razor code. Razor supports C# and Visual Basic programming languages.

3. How do you implement Forms authentication in MVC?

Create a MVC project.

Add these settings in web.config.

```
<authentication mode="Forms">
    <forms loginUrl="~/Login/Login" timeout="2880" />
</authentication>
<authorization>
    <allow users="*" />
    <deny users="?" />
</authorization>
```

Add a model for user credential structure like below:

```
public class User
{
    [Required]

    [RegularExpression("[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\\.[A-Za-z]{2,4}")]
    public string UserName { get; set; }
```

```

        [Required]
        public string passWord { get; set; }
    }

```

Create a Controller to handle Login event:

```

public class LoginController : Controller
{
    public ActionResult Login()
    {
        return View();
    }
}

```

//User is the model. and value for returnUrl will be automatically received by this action.

```

[HttpPost]
public ActionResult Login(User u, string returnUrl)
{
    if (ModelState.IsValid)
    {
        string struName = Convert.ToString(u.UserName).ToUpper().Trim();
        string strPassword =
Convert.ToString(u.passWord).ToUpper().Trim();

        //Verify credentials against database in real project
        if (struName == "123@123.com".ToUpper() && strPassword ==
"123".ToUpper())
        {

```

```

FormsAuthentication.SetAuthCookie(u.UserName.ToUpper(),
false);

if (!string.IsNullOrEmpty(returnUrl))
{
    return Redirect(returnUrl);
}
else
{
    return RedirectToAction("details", "Test");
}
}
else
{
    ModelState.AddModelError("authenticationError",
"User name or Password is wrong. Try it again");
    //lblError.Text = "User name or Password is wrong. Try it
again";
}
}
return View(u);
}

```

[Authorize]

[HttpGet]

public ActionResult logOut()

{

FormsAuthentication.SignOut();

return RedirectToAction("Login");

