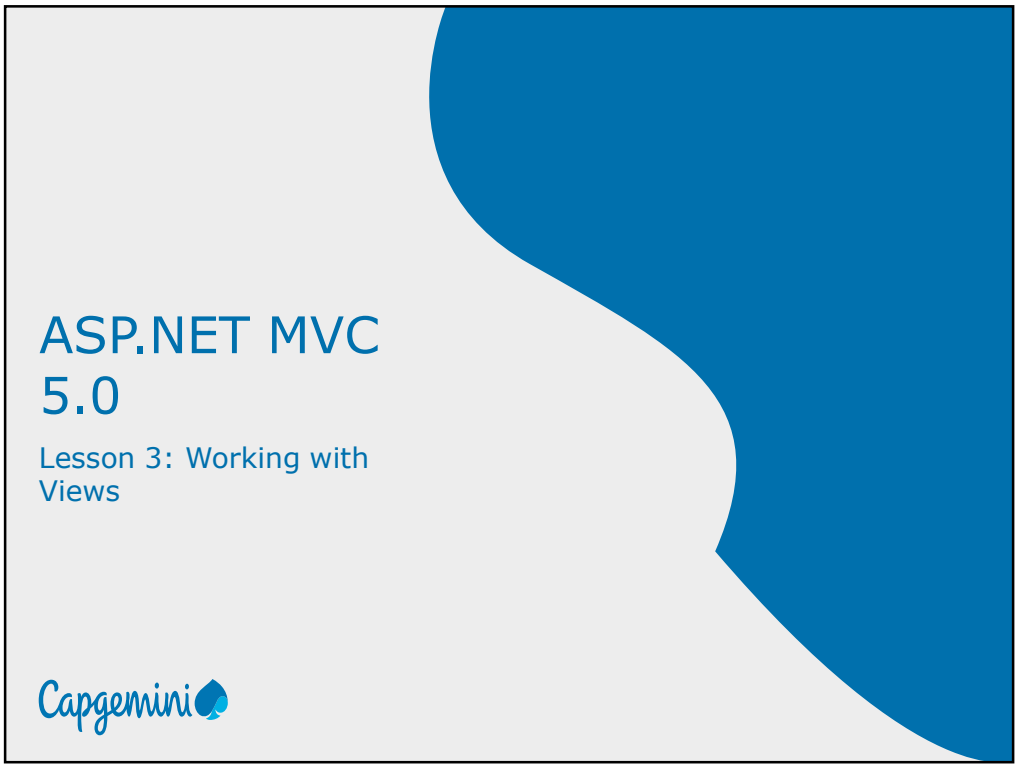

ASP.NET MVC
5.0
Lesson 3: Working with
Views
Capgemini

Lesson Objectives

- Razor syntax and keywords
- Using Layouts
- HTML Helpers
- Partial Views

2.1 ASP.NET MVC View Engines

View Engines

- View engine is used to render a page to the response.
- ASP.NET MVC supports two view engines which lets us to embed server-based code(C# and VB.NET) into WebPages.
  - ASPX view engine
  - Razor view engine

2.1 ASP.NET MVC View Engines

Razor view Engine

- Razor is based on ASP.NET and designed for creating web applications.
- It provides a simplified syntax.
- When server reads the page, it runs the razor code first before it sends the HTML page to the browser.
- ASP.NET MVC with Razor syntax will have the special file extension .cshtml(C#) or .vbhtml(VB)

2.1 ASP.NET MVC View Engines

Razor syntax

- Code : We can add code to a page using @ character
  - @
  - @( )
  - @{ }
- Content : Mixing Code and Plain Text
  - @:
  - <text></text>
- Comments : Server Side Comment
  - @* *@

Code Block :

```
@{
        int sampleInt = 34;
        string companyName = “Capgemini";
}
```

Access Variables :

```
@ companyName ;
```

Multi token Statements :

```
@("Your Company Name is : " + companyName);
```

Mixing code and plain text :

```
<text>Your company name is @companyName</text>
```

Or

```
@:Your company name is  @companyName
```

Comment :

```
@*
   Multi line Comment
*@
```

2.2 Working with Layout

Introduction to Layout

- Layout ensures to have a consistent look an feel across all of the pages with in the web site / web application.
- It is very similar to Master pages which is introduced in ASP.NET 2.0
- Razor view engine uses "layout pages" to define a common site template and then inherits its look and feel across all the views/ pages on the site.
- ASP.NET MVC 4 by default uses _Layout.cshtml as master page under the Shared folder.
- Default master page can be changed by changing the value of Layout variable in _ViewStart.cshtml

_ViewStart.cshtml

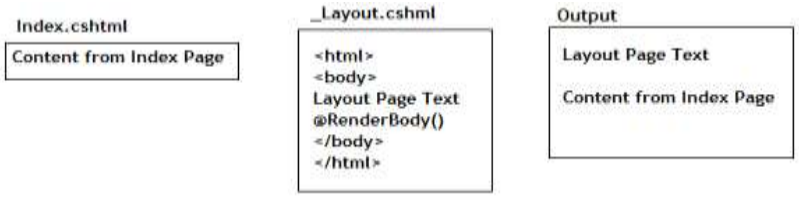
The _ViewStart.cshtml applied to all views in current folder and all sub folders i.e. While RazorViewEngine renders the view it executes _ViewStart.cshtml first and then renders the view.

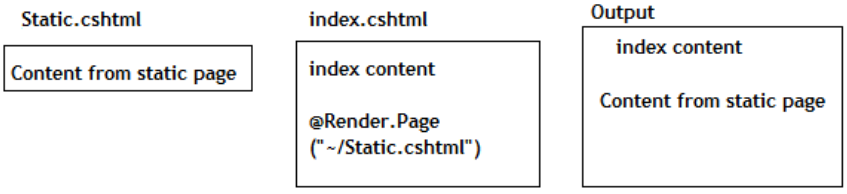
2.2 Working with Layout

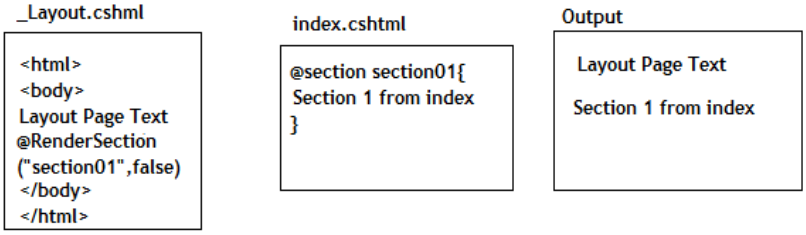
Render Body, Render Page & Render Section

- To Render the content between the master pages and content pages ASP.NET MVC offers 3 methods
  - @RenderBody() : It is used to render the portion of a content page inside the master page.
  - @RenderPage() : It is used to render the content of one page with in another page(layout or normal page)
  - @RenderSection() : It is used to render the content of a named section of content page inside the master page. This method take boolean value as second argument to refer the section is optional (false) or mandatory(true)

@RenderBody()
@RenderPage()
@RenderSection ()







2.3 HTML Helpers

HTML Helper

- An HTML Helper is just a method that returns a string having HTML or plain text and can be used in our Views.
- MVC includes standard helpers for the most common types of HTML elements, like HTML links and HTML form elements.
- Custom HTML Helpers can be created

As their name implies, HTML helpers help you work with HTML.
Forms are where most of the hard work happens inside an application, and are where you need to use HTML helpers the most.

In-Built HTML helpers to render HTML Form elements

· **BeginForm()**
· **CheckBox()**
· **DropDownList()**
· **EndForm()**
· **Hidden()**
· **ListBox()**
· **Password()**
· **RadioButton()**
· **TextArea()**
· **TextBox()**

HTML Helper to render readonly textbox
@Html.TextBox("txtEmployeeId", 101, new { @readonly="readonly"})
***//*Strongly typed**
@Html.TextBoxFor(employee => employee.Id)

2.4 Working with Partial Views

Partial View

- Partial view is used to define a view that will be rendered inside a parent view.
- Using partial view we can render a view inside a parental view and to create reusable content in the project.
- Partial views are implemented as ASP.NET user controls (.ascx)
- Partial view can access the parent view data
- Partial view plays a vital role in AJAX implementation(Partial page updation)

Custom Helper to remove the HTML tags from a string

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

namespace MyCustomHelper.Helpers
{
public static class MyHelper
{
  public static MvcHtmlString StripHTML(this HtmlHelper html, string input)
   {
    return new
MvcHtmlString(System.Text.RegularExpressions.Regex.Replace(input, "<.*?>",
string.Empty));
  }
 }
}
@using MyCustomHelper.Helpers
@Html.StripHTML("<div><p>Test</p></div>")
```

2.4 Working with Partial Views

Partial View

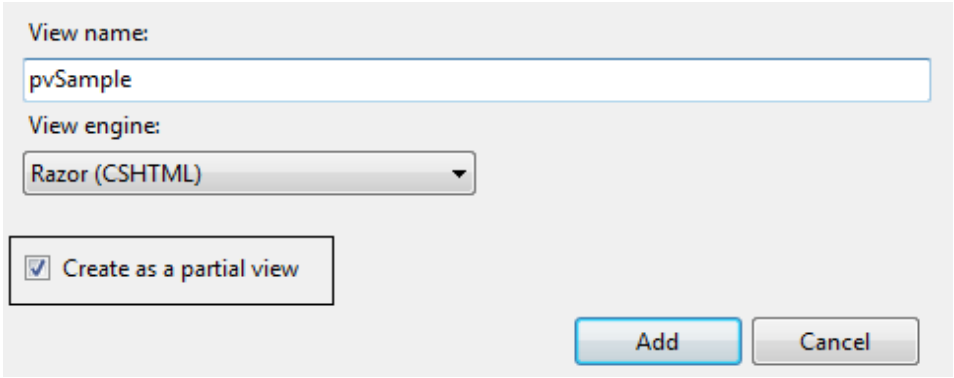
- Partial view is used to define a view that will be rendered inside a parent view.
- Using partial view we can render a view inside a parental view and to create reusable content in the project.
- Partial views are implemented as ASP.NET user controls (.ascx)
- Partial view can access the parent view data
- Partial view plays a vital role in AJAX implementation(Partial page updation)

Partial Views can be rendered using the following HTML Helpers

```
@Html.Partial("pvSample")

@{
   Html.RenderPartial("pvSample");
}
```

Creating View as a Partial View



DEMO

- View Demo

Summary

- View is responsible for rendering UI(HTML)
- Razor view engine provides a simplified syntax.
- Layout are very similar to Master pages.
- Most of the HTML helpers in the MVC framework exist to render small pieces of HTML
- Partial view is used to define a view that will be rendered inside a parent view.

Review Question

- Question 1: In a Razor view, to explicitly transition from C# code to literal text, you can use
  - A <text> tag
  - The @: delimiter

- Question 2: In a Razor layout view, you can place the content of the controller action's primary view by
  - Invoking RenderSection("Main")
  - Using a ContentPlaceHolder control
  - Invoking the RenderBody method
  - Invoking RenderSection("viewname")