

View Engine in ASP.NET MVC

[Home](#) > [Web Technologies](#) > [ASP.NET MVC](#)

« [Previous](#)

[Next](#) »

Tutorial

[ASP.NET MVC Home](#)

[MVC Introduction](#)

[MVC Environment Set Up](#)

[MVC Views](#)

[Model in MVC application](#)

[MVC Entity Framework](#)



View Engine renders the view into html form to the browser. By default, Asp.net MVC comes with two flavors Form (ASPX) and Razor View Engine. There are many types of view engines.

- 5) NDJango
- 6) Hasic
- 7) Brail

These engine works between your view and browser to provide HTML to your browser. ASPX View Engine is the default view engine for the ASP.NET MVC that comes with ASP.NET MVC from the start (MVC 1.0) but Razor View Engine is introduced with MVC3 or latter version.

- In Razor View Engine we use Layouts and in ASPX View Engine we use masterPages.
- In Razor View Engine we use PartialPage and in ASPX View Engine we use WebUserControls.
- Razor View Engine has .cshtml (with C#) and .vbhtml (with VB) extension for views, Layout and Partial views but in ASPX it uses same extension as in web application.
- Razor View Engine supports Test Driven Development but ASPX engine does not.
- Razor Engine prevents Cross-Site Scripting Attacks.

Razor Syntax in C#

Razor view engine uses @ character to start inline expressions, single statement blocks, and multi-statement blocks.

Related Topics

[HTML](#)[Apache HTTP server](#)[Ajax](#)

```
<body style="font-family:Arial; font-size:medium; color:darkblue">  
<h2>View Engin Demo</h2>  
<!-- Single statement blocks -->  
@{ var RollNo = 7; }  
@{ var StudentName = "Raj Parihar";  
<!-- Inline expressions -->  
<p>The student RollNo is:= @RollNo </p>  
<p>Name := @StudentName</p>  
<!-- Multi-statement block -->  
@{  
    var greeting = "Welcome to our site! Mr. " + @StudentName;  
    var Department = "Your department is IT";  
    var myMessage = greeting + " " + Department;  
}  
<p>The greeting is: @myMessage</p>
```

[XML](#)[jQuery](#)[HTML5](#)[ASP.NET MVC](#)[ASP.NET Practicals](#)[ASP.NET](#)

Execute the above program you will get the following output.

View Engin Demo

The student RollNo is:= 7

Name := Raj Parihar

The greeting is: Welcome to our site! Mr. Raj Parihar Your department is IT

```
@if (Model != null)
{
    foreach (var item in Model)
    {
        <ul>
            <li>
                @Html.DisplayFor(modelItem => item.Name)
            </li>
            <li>
                @Html.DisplayFor(modelItem => item.Address)
            </li>
            <li>
                @Html.DisplayFor(modelItem => item.City)
                <br />
            </li>
        </ul>
    }
}
```

ASPX view

This is the default view engine for MVC 1 and MVC 2. ASPX View Engine is also known as **Web Form View Engine** and its syntax for rendering server-side contents are as "<%= %>" or "<=: %>"

```
string strVariable = "Hello World !!";  
bool boolVar = true;  
string[] arrVar = new string[] { "A", "B", "C"};  
%>
```

Looping statement in ASPX view engine

```
<ul>  
  <%foreach (var item in ListItem)  
    { %>  
    <li><%=item%></li>  
    <%}  
  %>  
</ul>
```

Example:

Let take an example that have hyperlink and textbox control in aspx view.
Create a view and update the code as follows.

```
<%@ Page Language="C#"  
Inherits="System.Web.Mvc.ViewPage<dynamic>" %>  
<!DOCTYPE html>
```

```
</head>
<body>
  <div>
    <p>
      To learn more MVC, click the following link:
      <%= Html.ActionLink("About this Website", "Home" ) %>
      <br/><br/>
      <label for="FirstName">First Name:</label>
      <%= Html.TextBox("FirstName") %>
      <%= Html.ValidationMessage("FirstName", "*") %>
    </p>
  </div>
</body>
</html>
```

Output:

To learn more MVC, click the following link: [About this Website](#)

First Name: *

Difference between ASPX and Razor View Engine

ASPX View Engine	RAZOR View Engine
System.Web.Mvc.WebFormViewEngine	System.Web.Razor is the namespace for

.aspx is file extension for web page and .ascx is for user controls	.cshtml (with C#) and .vbhtml (with VB) extension for views, Layout and Partial views
No support for TDD (Test Driven Development).	Supports TDD (Test Driven Development).
ASPX View Engine is comparatively fast.	It is slow as compared to ASPX View Engine.
No support for Cross-Site Scripting attacks.	Supports for Cross-Site Scripting attacks.

[« Previous](#)[Next »](#)

SPONSORED SEARCHES

[engine in mvc](#)[simple example of mvc in asp net c#](#)[asp net mvc engines support](#)[c# and mvc](#)

