# How to create and access session variables in ASP.NET MVC (ASPNETMVCSession)

## Introduction

This sample demonstrates how to create and access session variables in ASP.NET MVC. In this sample, we will demo two ways to achieve this. One is to directly access **HttpContext.Current.Session**. The other is to create an extension method for **HttpContextBase**. We type some words selected from the page and submit it to the controller. In the controller, we save the text to session.

**Building the Sample**

For this sample to run, you must install the ASP.NET MVC 4 and if you use Visual Studio 2010, you must install Visual Studio 2010 SP1.

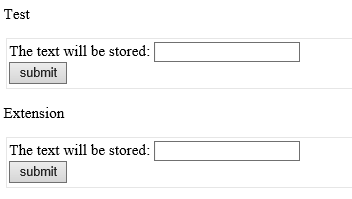
* <http://www.asp.net/mvc/mvc4>
* [ASP.NET MVC 4 for Visual Studio 2010 SP1](http://www.microsoft.com/web/handlers/webpi.ashx?command=getinstallerredirect&appid=MVC4VS2010_Loc)

## Running the Sample

1. Open the CSASPNETMVCSession.sln/VBASPNETMVCSession.sln.
2. Press **F5** or **Ctrl + F5** to run the application. You will see a page as shown below:



1. Type some text to the text box and then click the submit button. Then the page will be displayed as follows (Each submit button works for the text box just above it):



1. Validation finished.

## Using the Code

1. Create an **ASP.NET MVC 4 Web Application** project in **Visual Studio**.

We can use the default **template** and default “**View engine**”. In the demo project, our choice is the **Empty** template.

1. In the **Solution Explorer**, browse to the **Controllers** folders and then double-click to add a class. You can name it as **HomeController.**
2. We will save and access the session in the **HomeController** class. We will use two ways to achieve this.

Using HttpContext.Current.Session:

1. Save the session variables:

|  |
| --- |
| -Code block start-  --C# code snippet start--  System.Web.HttpContext.Current.Session["sessionString"] = sessionValue;  --C# code snippet end--  --VB code snippet start--  System.Web.HttpContext.Current.Session("sessionString") = sessionValue  --VB code snippet end--  -Code block end- |

1. Access the session variables:

|  |
| --- |
| -Code block start-  --VB code snippet start--  ViewData("sessionString") = TryCast(System.Web.HttpContext.Current.Session("sessionString"), [String])  --VB code snippet end--  --C# code snippet start--  ViewData["sessionString"] = System.Web.HttpContext.Current.Session["sessionString"] as String;  --C# code snippet end--  -Code block end- |

Creating an extension method for HttpSessionStateBase:

1. Firstly, write extension method to the HttpSessionStateBase. The code snippet resembles the following:

|  |
| --- |
| -Code block start-  --C# code snippet start--  public static class SessionExtensions  {  /// <summary>  /// Get value.  /// </summary>  /// <typeparam name="T"></typeparam>  /// <param name="session"></param>  /// <param name="key"></param>  /// <returns></returns>  public static T GetDataFromSession<T>(this HttpSessionStateBase session, string key)  {  return (T)session[key];  }  /// <summary>  /// Set value.  /// </summary>  /// <typeparam name="T"></typeparam>  /// <param name="session"></param>  /// <param name="key"></param>  /// <param name="value"></param>  public static void SetDataToSession<T>(this HttpSessionStateBase session, string key, object value)  {  session[key] = value;  }  }  --C# code snippet end--  --VB code snippet start--  Public Module SessionExtensions  Sub New()  End Sub  ''' <summary>  ''' Get value.  ''' </summary>  ''' <typeparam name="T"></typeparam>  ''' <param name="session"></param>  ''' <param name="key"></param>  ''' <returns></returns>  <System.Runtime.CompilerServices.Extension> \_  Public Function GetDataFromSession(Of T)(session As HttpSessionStateBase, key As String) As T  Return DirectCast(session(key), T)  End Function  ''' <summary>  ''' Set value.  ''' </summary>  ''' <typeparam name="T"></typeparam>  ''' <param name="session"></param>  ''' <param name="key"></param>  ''' <param name="value"></param>  <System.Runtime.CompilerServices.Extension> \_  Public Sub SetDataToSession(Of T)(session As HttpSessionStateBase, key As String, value As Object)  session(key) = value  End Sub  End Module  --VB code snippet end--  -Code block end- |

1. Save the session variables by using the extension method:

|  |
| --- |
| -Code block start-  --VB code snippet start--  Session.SetDataToSession<string>("key1", sessionValue);  --VB code snippet end--  --VB code snippet start--  Session.SetDataToSession(Of String)("key1", sessionValue)  --VB code snippet end--  -Code block end- |

1. Access the session variables by using the extension method:

|  |
| --- |
| -Code block start-  --C# code snippet start--  string value = Session.GetDataFromSession<string>("key1");  ViewData["sessionStringByExtensions"] = value;  --C# code snippet end--  --VB code snippet start--  Dim value As String = Session.GetDataFromSession(Of String)("key1")  ViewData("sessionStringByExtensions") = value  --VB code snippet end--  -Code block end- |

1. The html code snippet resembles the following:

|  |
| --- |
| -Code block start-  --HTML code snippet start--  <body>  <div>  <%Using (Html.BeginForm("SaveSession", "Home", FormMethod.Post))  %>  <p><%=ViewData("sessionString")%> </p>  <fieldset>  <div>  The text will be stored: <%=Html.TextBox("sessionValue")%>  </div>  <div>  <input type="submit" id="submitToMethod1" name="submit" value="submit" />  </div>  </fieldset>  <%  End Using  %>  <%Using (Html.BeginForm("SaveSessionByExtensions", "Home", FormMethod.Post))  %>  <p>  <%=ViewData("sessionStringByExtensions")%>  </p>  <fieldset>  <div>  The text will be stored: <%=Html.TextBox("sessionValue")%>  </div>  <div>  <input type="submit" id="submitToMethod2" name="submit" value="submit" />  </div>  </fieldset>  <%  End Using  %>  </div>  </body>  --HTML code snippet end--  --VB code snippet start--  <body>  <div>  @Using (Html.BeginForm("SaveSession", "Home", FormMethod.Post))  @<p>  @ViewData("sessionString")  </p>  @<fieldset>  <div>  The text will be stored: @Html.TextBox("sessionValue")  </div>  <div>  <input type="submit" id="submitToMethod1" name="submit" value="submit" />  </div>  </fieldset>  End Using  @Using (Html.BeginForm("SaveSessionByExtensions", "Home", FormMethod.Post))  @<p>  @ViewData("sessionStringByExtensions")  </p>  @<fieldset>  <div>  The text will be stored: @Html.TextBox("sessionValue")  </div>  <div>  <input type="submit" id="submitToMethod2" name="submit" value="submit" />  </div>  </fieldset>  End Using  </div>  </body>  --VB code snippet end--  --C# code snippet start--  <body>  <div>  @using (Html.BeginForm("SaveSession", "Home", FormMethod.Post))  {  <p>@ViewData["sessionString"] </p>  <fieldset>  <div>  The text will be stored: @Html.TextBox("sessionValue")  </div>  <div>  <input type="submit" id="submitToMethod1" name="submit" value="submit" />  </div>  </fieldset>  }  @using (Html.BeginForm("SaveSessionByExtensions", "Home", FormMethod.Post))  {  <p>@ViewData["sessionStringByExtensions"]</p>  <fieldset>  <div>  The text will be stored: @Html.TextBox("sessionValue")  </div>  <div>  <input type="submit" id="submitToMethod2" name="submit" value="submit" />  </div>  </fieldset>  }  </div>  </body>  --C# code snippet end--  -Code block end- |

1. We create an enum named AllViewsNames to switch the engine mode between Razor and ASPX.

|  |
| --- |
| -Code block start-  --C# code snippet start--  #region All Views' names  enum AllViewsNames  {  RazorIndex,  ASPXIndex  }  #endregion  static AllViewsNames currentViewEnum = AllViewsNames.ASPXIndex;  // Current view name;  string strCurrentView = currentViewEnum == AllViewsNames.RazorIndex ? "Index" : "TestPage";  --C# code snippet end--  Insert other Programming Language Code Snippet here  -Code block end- |

1. You can run and debug it.

## More Information

HttpSessionStateBase Class  
<http://msdn.microsoft.com/en-us/library/system.web.httpsessionstatebase.aspx>  
HttpContext Class  
<http://msdn.microsoft.com/en-us/library/system.web.httpcontext(v=vs.110).aspx>  
HttpContext.Current Property  
<http://msdn.microsoft.com/en-us/library/system.web.httpcontext.current(v=vs.110).aspx>  
HttpContext.Session Property  
<http://msdn.microsoft.com/en-us/library/vstudio/system.web.httpcontext.session(v=vs.100).aspx>  
Controller.RedirectToAction Method  
<http://msdn.microsoft.com/en-us/library/system.web.mvc.controller.redirecttoaction(v=vs.118).aspx>  
FormExtensions.BeginForm Method  
<http://msdn.microsoft.com/en-us/library/system.web.mvc.html.formextensions.beginform(v=vs.118).aspx>