# How To Disable ASP Session State in ASP.NET and in MVC4

We can disable the Session foe whole application in Web.config

**Disable Session State at the Application Level**

The following steps demonstrate how to disable session state at the Application level, which affects all pages in the application:

1. Start Microsoft Visual Studio .NET, and create a new ASP.NET Web Application.
2. In Solution Explorer, double-click **Web.config** to view the contents of this file.
3. Locate the **<sessionState>** section, and set the **mode** value to **Off**.
4. Save the file and/or the project to disable session state throughout all pages in the application.

```
<sessionState mode="Off" customProvider="DefaultSessionProvider">
```

**Disable Session State at the Page Level**

1. At the top of the page, add **EnableSessionState="false"** in the **@ Page** directive. The modified attribute should appear similar to the following

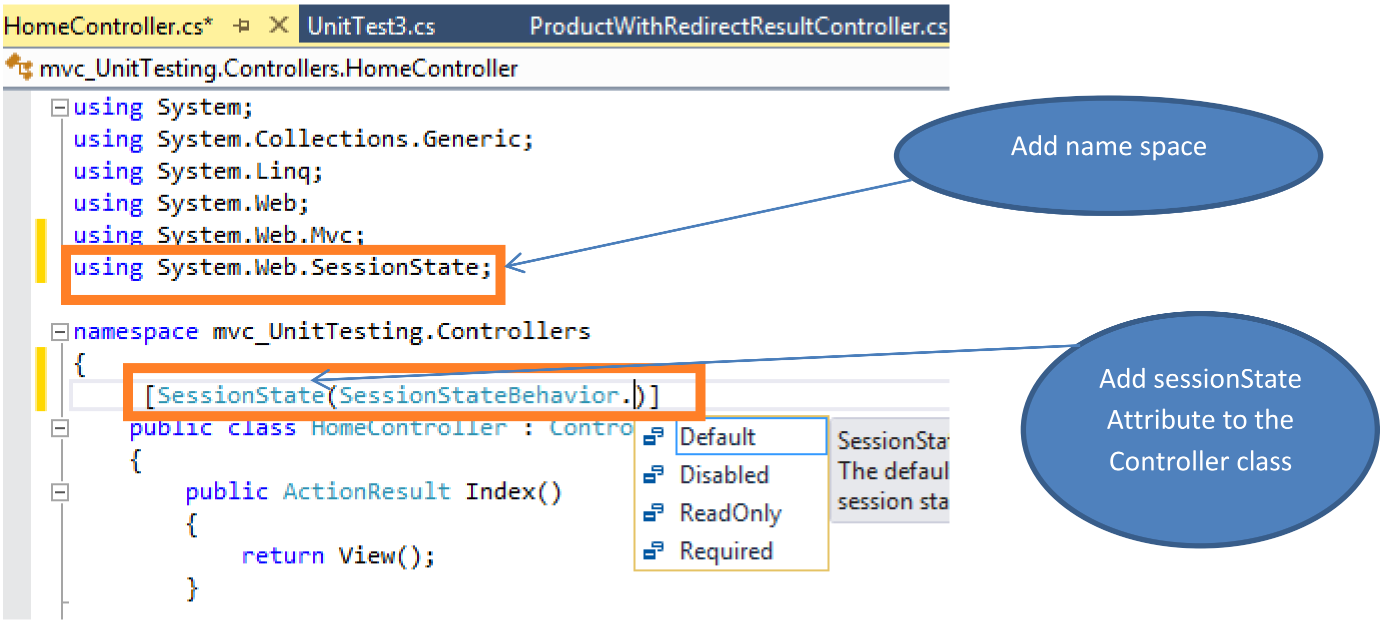
For a particular lavel

```
<%@ Page Title="Home Page" Language="C#" MasterPageFile="~/Site.Master"
AutoEventWireup="true" CodeBehind="Default.aspx.cs"
Inherits="WebApplication9._Default" EnableSessionState="False" %>
```

# Controlling Session Behavior in Asp.Net MVC4

By default**, Asp.Net MVC support session state**.

Session is used to store data values across requests. Whether you store some data values with in the session or not Asp.Net MVC must manage the session state for all the controllers in your application that is time consuming. Since, **session is stored on server side** and consumes server memory, hence **it also affect your application performance**.

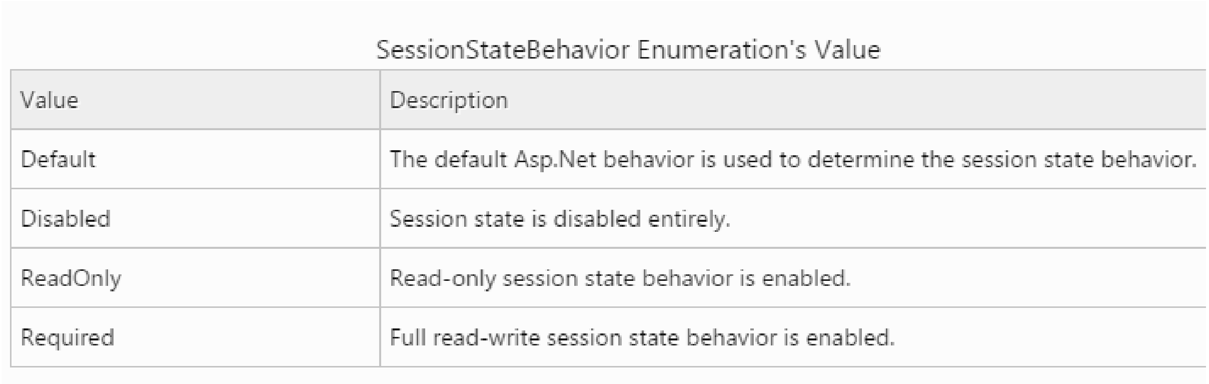


```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using System.Web.SessionState;

namespace mvc_UnitTesting.Controllers
{
    [SessionState(SessionStateBehavior.)]
    public class HomeController : Contro
    {
        public ActionResult Index()
        {
            return View();
        }
```

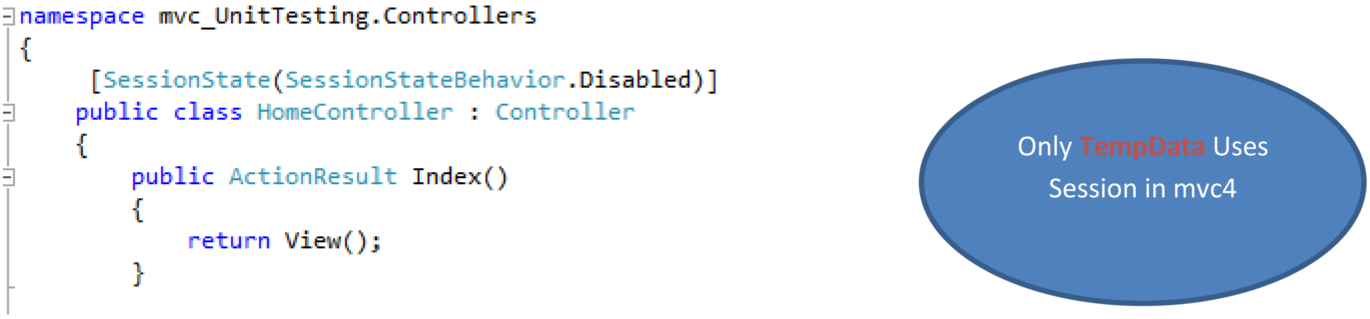
Add name space

Add sessionState Attribute to the Controller class



SessionStateBehavior Enumeration's Value

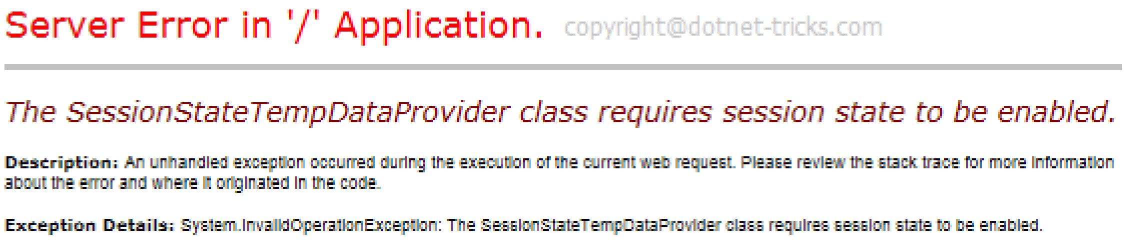
| Value | Description |
|---|---|
| Default | The default Asp.Net behavior is used to determine the session state behavior. |
| Disabled | Session state is disabled entirely. |
| ReadOnly | Read-only session state behavior is enabled. |
| Required | Full read-write session state behavior is enabled. |



```
namespace mvc_UnitTesting.Controllers
{
    [SessionState(SessionStateBehavior.Disabled)]
    public class HomeController : Controller
    {
        public ActionResult Index()
        {
            return View();
        }
```

Only TempData Uses Session in mvc4

n Asp.Net MVC, **TempData** use **session state** for storing the data values across requests. Hence, when you will disabled the session state for the controller, it will throw the exception as shown below:



Server Error in '/' Application. copyright@dotnet-tricks.com

*The SessionStateTempDataProvider class requires session state to be enabled.*

**Description:** An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

**Exception Details:** System.InvalidOperationException: The SessionStateTempDataProvider class requires session state to be enabled.

# If Session state disabled in web config what is the impact on ViewBag, ViewData and Tempdata in MVC

**Only `TempData` is affected** (as it relies on `Session`).

```
stringstr = TempData[“MyData”];
```

```
@TempData[“MyData”];
```

Condition 3 (Read and Keep) :- If you read the “TempData” and call the “Keep” method it will be persisted.

Hide Copy Code

```
@TempData[“MyData”];
TempData.Keep(“MyData”);
```

Condition 4 ( Peek and Read) :- If you read “TempData” by using the “Peek” method it will persist for the next request.

Hide Copy Code

```
stringstr = TempData.Peek("Td").ToString();
```