ASP.NET MVC - TempData

TempData in ASP.NET MVC can be used to store temporary data which can be used in the subsequent request. TempData will be cleared out after the completion of a subsequent request.

TempData is useful when you want to transfer non-sensitive data from one action method to another action method of the same .It is dictionary type which is derived from [TempDataDictionary](https://msdn.microsoft.com/en-us/library/system.web.mvc.tempdatadictionary(v=vs.118).aspx" \t "_blank).

You can add a key-value pair in TempData as shown in the below example.

Example: TempData

public class HomeController : Controller

{

// GET: Student

public HomeController()

{

}

public ActionResult Index()

{

TempData["name"] = "Test data";

TempData["age"] = 30;

return View();

}

public ActionResult About()

{

string userName;

int userAge;

if(TempData.ContainsKey("name"))

userName = TempData["name"].ToString();

if(TempData.ContainsKey("age"))

userAge = int.Parse(TempData["age"].ToString());

// do something with userName or userAge here

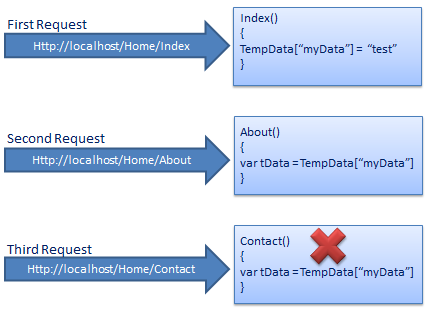
return View();

}

}

In the above example, we have added data into TempData and accessed the same data using a key inside another action method. Please notice that we have converted values into the appropriate type.

The following figure illustrates TempData.

TempData

TempData internally uses session to store the data. So the data must be serialized if you decide you to switch away from the default Session-State Mode, and use State Server Mode or SQL Server Mode.

As you can see in the above example, we add test data in TempData in the first request and in the second subsequent request we access test data from TempData which we stored in the first request. However, you can't get the same data in the third request because TempData will be cleared out after second request.

Call TempData.Keep() to retain TempData values in a third consecutive request.

Example: TempData.Keep()

public class HomeController : Controller

{

public HomeController()

{

}

public ActionResult Index()

{

TempData["myData"] = "Test data";

return View();

}

public ActionResult About()

{

string data;

if(TempData["myData"] != null)

data = TempData["myData"] as string;

TempData.Keep();

return View();

}

public ActionResult Contact()

{

string data;

if(TempData["myData"] != null)

data = TempData["myData"] as string;

return View();

}

}

 Points to Remember :

1. TempData can be used to store data between two consecutive requests. TempData values will be retained during redirection.
2. TemData is a TempDataDictionary type.
3. TempData internaly use Session to store the data. So think of it as a short lived session.
4. TempData value must be type cast before use. Check for null values to avoid runtime error.
5. TempData can be used to store only one time messages like error messages, validation messages.
6. Call TempData.Keep() to keep all the values of TempData in a third request.