

**The results are in!** See what nearly 90,000 developers picked as their most loved, dreaded, and desired coding languages and more in the 2019 Developer Survey.

## Access ViewState from Static method in aspx page

[Ask Question](#)

▲  
7 suppose i have one static method and i need to access viewstate from that method...how could i do so...i know it is not possible but there must be some way out.

▼  
★  
1

```
[WebMethod]
public static string GetData(int CustomerID)
{
    string outputToReturn = "";
    ViewState["MyVal"]="Hello";
    return outputToReturn;
}
```

[asp.net](#)

edited Sep 5 '11 at 14:29



[ROMANARMY](#)

13.9k 2 48 76

asked Sep 5 '11 at 14:00



[Thomas](#)

14.3k 96 305 533

2 If you know it's not possible, why do you think there must be a way to do so? – [ROMANARMY](#) Sep 5 '11 at 14:30

1 The ViewState of what? ViewState is part of a Control/Page - but you appear to be calling a (static?) WebService , so there is no ViewState . – [TheCodeKing](#) Sep 5 '11 at 14:39

## 3 Answers



14



You can get the reference to the page via [HttpContext.CurrentHandler](#). But since [Control.ViewState](#) is protected **you can't access it** (without using reflection) as opposed to the `Session` which is accessible via `HttpContext.Current.Session`.

So either don't use a static method, use the `Session` or use this reflection approach:

```
public static string CustomerId
{
    get { return (string)GetCurrentPageViewState()["CustomerId"]; }
    set { GetCurrentPageViewState()["CustomerId"] = value; }
}

public static System.Web.UI.StateBag GetCurrentPageViewState()
{
    Page page = HttpContext.Current.Handler as Page;
    var viewStateProp = page?.GetType().GetProperty("ViewState",
        BindingFlags.FlattenHierarchy |
        BindingFlags.Instance |
        BindingFlags.NonPublic);
    return (System.Web.UI.StateBag) viewStateProp?.GetValue(page);
}
```

However, this won't work if called via Webservice, because then it's outside of [Page-Lifecycle](#).

edited Aug 26 '16 at 8:10

answered Sep 5 '11 at 14:36



Rango

367k 46 475 739

Home

PUBLIC

Stack Overflow

Tags

Users

Jobs

Teams

Q&A for work



[Learn More](#)

thanks...i have one static method in aspx page. – [Thomas](#) Sep 5 '11 at 17:16

4 ViewState is a protected field so this doesn't work – [Tudor Carean](#) Aug 21 '13 at 11:53

I tried this and worked for me:

2

- Create a class containing the properties of the ViewState you want to access to
- In the constructor pass the real ViewState
- Create a static instance of the class but not initialize it
- In the PageLoad initialize Not static class and the static one
- Access the ViewState using static class properties

-

```
public class Repository
{
    public int a
    {
        get
        {
            if (_viewState["a"] == null)
            {
                return null;
            }

            return (int)_viewState["a"];
        }
        set
        {
            _viewState["ActiveGv"] = value;
        }
    }

    public StateBag _viewState;
```

```
public Repository(StateBag viewState)
{
    _viewState = viewState;
}

static Repository staticRepo;

protected void Page_Load(object sender, EventArgs e)
{
    Repository repo = new Repository(ViewState);

    staticRepo = repo;
}

public static void testMethod()
{
    int b = staticRepo.a;
}
```

answered May 31 '17 at 13:49

[Ders](#)

31 4

10

You might be able use `[WebMethod(EnableSession=true)]` for your `PageMethod`, and use `Session` instead of `ViewState`. Remember, with a static `PageMethod` no instance of the `Page` class is ever created, so nice things like `ViewState` simply are not there and there is no way to make them be there.

edited Jul 8 '13 at 9:34

[Bishan](#)

7,219 42 131 208

answered Sep 5 '11 at 17:40

[Peter Bromberg](#)

1,387 7 11

