

How to set values to ViewBag in ActionFilterAttribute ASP MVC 5?

Asked 5 years, 8 months ago Active 3 years ago Viewed 21k times



28



1



Hello I would like to create my custom ActionFilterAttribute for each controller in my application, this attribute should set some ViewBag values. Is ActionFilterAttribute would be fine for it and how to get access to viewbag in ActionFilterAttribute ?

[asp.net-mvc](#)

[asp.net-mvc-5](#)

asked Jun 25 '14 at 12:16



[Arbejdsglæde](#)

10.8k 17 63 127

3 Answers



66



You can do like this

```
public class SomeMsgAttribute : FilterAttribute, IResultFilter
{
    public void OnResultExecuted(ResultExecutedContext filterContext)
    {
    }

    public void OnResultExecuting(ResultExecutingContext filterContext)
    {
        filterContext.Controller.ViewBag.Msg= "Hello";
    }
}
```

Using:

```
[SomeMsg]
public ActionResult Index()
{
    return View();
}
```

answered Jun 25 '14 at 12:23



Ilya

6,204 2 34 60

Do you have any explanations about why this works in `Executing` but not in `Executed` ? – [Tolga Evcimen](#) Mar 11 '16 at 7:54

2 @TolgaEvcimen `OnResultExecuted` is invoked after the razor view is rendered. By that time, it is too late to change the output. – [DPac](#) Jun 22 '16 at 19:58

I see, I always thought it as `OnResultExecuted` invoked immediately after the action method returned, pity :) I should've thought it :) Thanks. – [Tolga Evcimen](#) Jun 23 '16 at 8:37

try this

12

```
public class CustomFilterAttribute : ActionFilterAttribute
{
    public override void
    OnActionExecuting(ActionExecutingContext filterContext)
    {
        // get the view bag

        var viewBag = filterContext.Controller.ViewBag;

        // set the viewbag values
        viewBag.CustomValue = "CustomValue";
    }
}
```

answered Jun 25 '14 at 12:23

[Anto Subash](#)

2,630 2 20 28

▲ To transfer data from a different controller action

1

```
public override void OnActionExecuting(ActionExecutingContext filterContext)
{
    EmployeeTrackingSystemAndMISEntities db = new
    EmployeeTrackingSystemAndMISEntities();

    var UserCookie = filterContext.HttpContext.Request.Cookies["UserUniqueID"];

    RouteValueDictionary redirectTargetDictionary = new RouteValueDictionary();
    redirectTargetDictionary.Add("action", "UserLogIn");
    redirectTargetDictionary.Add("controller", "Login");

    var TempData = filterContext.Controller.TempData;
    TempData["Status"] = "Please log in as Admin";

    filterContext.Result = new RedirectToRouteResult(redirectTargetDictionary);
}
```

answered Feb 14 '17 at 10:51

[Arun Prasad E S](#)

5,571 4 53 59