Get current System.Web.UI.Page from HttpContext?

Ask Question

83

17

This is actually a two part question. First,does the HttpContext.Current correspond to the current System.UI.Page object?

And the second question, which is probably related to the first, is why can't I use the following to see if the current page implements an interface:

```
private IWebBase FindWebBase()
{
    if (HttpContext.Current as IWebBase != null)
    {
        return (IWebBase)HttpContext.Current.;
    }
    throw new NotImplementedException("Crawling for IWebBase not implemented yet");
}
```

The general context is that some controls need to know whether they are executing as a SharePoint webpart, or as part of an Asp.Net framework.

I have solved the problem by requiring the control to pass a reference to itself, and checking the Page property of the control, but I'm still curious why the above does not work.

The compiler error is: Cannot convert System.Web.HttpContext to ...IWebBase via a reference conversion, boxing conversion, unboxing conversion, wrapping conversion or null type conversion.

c# asp.net httpcontext

edited Apr 7 '12 at 0:01
user166390

asked Sep 12 '08 at 1:37

Travis
7,298 2 24 39

4 Answers

140

No, from MSDN on HttpContext.Current: "Gets or sets the HttpContext object for the current HTTP request."

In other words it is an HttpContext object, not a Page.

You can get to the Page object via HttpContext using:

```
Page page = HttpContext.Current.Handler as Page;

if (page != null)
{
     // Use page instance.
}
```

edited Feb 9 '10 at 8:11

answered Sep 12 '08 at 1:42

Ash
47.3k 29 140 165

just what I need... thanks... :) – AceMark Jan 11 '10 at 2:52

This gave me the access I needed inside an assembly to Session, Request, and Response. Thank you. – froggythefrog Mar 3 '13 at 1:57

Thankx , Nice information. It helped me. – Ratna Mar 16 '13 at 6:35

2 Down-Voted because incorrect. The answer with HttpContext.Current.CurrentHandler is correct! If you do a Server.Transfer, HttpContext.Current.Handler WILL BE THE PREVIOUS PAGE, HttpContext.Current.CurrentHandler WILL BE THE CURRENT PAGE – mike Jun 8 '16 at 15:09

37

You're looking for `HttpContext.Handler`. Since Page implements IHttpHandler, you'll obtain a reference to the currently executing page.You'll have to cast it, or at least try to cast it to the particular type you're looking for.

`HttpContext.Current` simply returns the singleton instance of HttpContext. Therefore, it is not and can never be, a page.

edited Jun 20 '12 at 10:42

alony
9,605 3 31 46

answered Sep 12 '08 at 1:42

Kilhoffer
20.9k 20 89 118

2 Just a note to anyone reading this. The answer below is the same but with an example (ie you use HttpContext.Current.Handler). – mike nelson Oct 11 '09 at 20:56

15

You may want to use `HttpContext.Current.CurrentHandler` if you want the precise page that is currently executing. For example, a request for Default.aspx is sent, but an error is thrown and you do a `Response.Transfer` to your custom ErrorHandler.aspx page. `CurrentHandler` will return the instance of ErrorHandler.aspx (if called

after the error) whereas `HttpContext.Current.Handler` would return an instance of Default.aspx.

edited May 15 '12 at 18:17

community wiki
3 revs, 2 users 75%
user452427

0

Please see my answer :
[Why HttpContext.Current.Handler is null?]

Maybe resolved your problem.

edited May 23 '17 at 12:09
Community ♦
1 1

answered Aug 1 '13 at 0:44

Amin Ghaderi
758 8 16

-1 this is the same as the accepted answer. – John Saunders Aug 1 '13 at 0:53

I'm using the same code for my work , but I was so annoyed because I was not aware of the details of work details. So developed this code for my worke . I thought that Put code Here To help friends that have my problem. Thank John Saunders. – Amin Ghaderi Aug 1 '13 at 1:15