Is there a way to clear query string parameters when posting back?

Ask Question



11

I have a form that sometimes gets linked to with some query string parameters. The problem is that when I post back the form, the query string parameter is still there. It's not really an issue the way I have it setup, but I just don't like it being there, and could see it being a problem if you needed to check for input in a certain order.



Is there a way to clear that query string parameter in an easy, clean way? I know I could change the PostBackURL on the button, but that doesn't seem too efficient.

asp.net postback query-string postbackurl

edited Nov 25 '10 at 4:12



Alex Angas **34.5k** 37 123 197

asked May 12 '09 at 17:26



John Bubriski ♦

9.785 29 109

11 Answers



No, I haven't seen a way to clear it out without a redirect.





1,348 1 10 14



1 same here . – Greg B May 12 '09 at 17:30

Is there a clean way to do a Response.Redirect() to the page you're currently on? – John Bubriski ♦ May 12 '09 at 18:08

3 Try: Response.Redirect(Request.CurrentExecutionFilePath()) - rvarcher May 12 '09 at 19:03

I know this is really old, but CurrentExecutionFilePath is a property, so you shouldn't have the parentheses at the end. Maybe consider adding your comment to your answer to make it more complete? – Zack Aug 19 '14 at 13:11



I think you've answered your own question. Use the PostBackURL property.





<asp:Button PostBackUrl='<%# Request.ServerVariables["URL"]
Text="Submit" />

Or something like

```
foreach (Control ctrl in Page.Controls)
{
    if (ctrl is Button)
    {
        ((Button)ctrl).PostBackUrl = Request.ServerVariable)
    }
}
```

answered May 12 '09 at 19:10





put this at the bottom of your page?

```
<script language="javascript" type="text/javascript">
   document.forms[0].action = window.location.pathname;
</script>
```

answered May 12 '09 at 17:38



Or from code-behind: form1.Action = Request.Url.GetLeftPart(UriPartial.Path); - user764754 Jul 9 '14 at 8:27



I was stuck with same issue.

I solved it using the following code block:



```
private void ClearQueryString()
    PropertyInfo isreadonly =
typeof(System.Collections.Specialized.NameValueCollection)
BindingFlags.Instance | BindingFlags.NonPublic);
   isreadonly.SetValue(this.Request.QueryString, false, n
   this.Request.QueryString.Remove("Action");
   this.Request.QueryString.Remove("param1");
   this.Request.QueryString.Remove("paramN");
```

edited Jun 2 '14 at 10:20



answered Jun 2 '14 at 10:08





You can in some cases use the <code>Server.Transfer()</code> method, which has an overload that allows you to specify whether the querystring and form data should be preserved or not.



answered May 12 '09 at 17:30



janhartmann 9,402 12 67 121



I assume that you can't rely on Page.IsPostBack for some reason?

3



If what you're doing is server-side, then it's simple to add a check for IsPostBack in your methods (Page_Load, OnInit, etc) and only processing the querystrings if it's not a post back (i.e. the initial request).

answered May 12 '09 at 20:20



Zhaph - Ben Duguid 24.3k 4 65 107



I just came across the same problem and after some hunting on the Web I found this snippet:

3



public void ClearQueryStrings()
{
 string clientCommand = string.Format()

Home

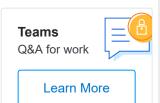
PUBLIC

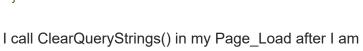


Tags

Users

Jobs





ClientScript.RegisterStartupScript(this.GetType(), "qr

I call ClearQueryStrings() in my Page_Load after I am done processing the original query string parameters. When the page posts back, the parameters are gone.

Original article is here.

edited Aug 27 '15 at 20:11



afuzzyllama **5,572** 4 38 56

answered Jan 20 '10 at 15:22



Miguel



using System.Collections;
using System.Reflection;





Use the following code to Clear Query String Parameters.

```
// To clear Query string value
PropertyInfo isreadonly =
typeof(System.Collections.Specialized.NameValueCollection)
BindingFlags.Instance | BindingFlags.NonPublic);

// make collection editable
isreadonly.SetValue(this.Request.QueryString, false, null)

// remove
this.Request.QueryString.Remove("YourQueryStringParameter"
```

edited Feb 18 '13 at 16:31

answered Feb 18 '13 at 16:07





This method is not what you would call a redirect, but it definitely will erase the query string. Simply add this line to the end of your JavaScript code:



```
window.location.search = "";
```

Or

Add this to the end of your HTML page:

```
<script type="text/javascript">
    window.location.search = "";
</script>
```

Depending on where you insert the code, the timing of erasing the query string will change.

answered Jun 23 '14 at 19:57





0

```
PropertyInfo isreadonly =
typeof(System.Collections.Specialized.NameValueCollection)
BindingFlags.Instance | BindingFlags.NonPublic);
```

this.Request.QueryString.Remove("status");

answered Mar 21 '16 at 7:49



Isn't this what most of the other answers are already suggesting? - Wai Ha Lee Mar 21 '16 at 8:07



You Can try

Response.Redirect(Request.CurrentExecutionFilePath()); Its working for me.



answered May 17 '16 at 10:27

