

microsoft.public.dotnet.framework.aspnet: [ENTER] key causing major headaches.

## [ENTER] key causing major headaches.

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.aspnet/2004-06/6057.html>

---

*From:* Eric (*ejproductions-msnews7831\_at\_mailblocks.com*)

*Date:* 06/25/04

Date: Fri, 25 Jun 2004 10:06:30 -0700

Hi All,

I'm very experienced in traditional ASP and am new to (am learning) ASP.NET. FYI: I am initially learning ASP.NET with VB.NET to ease the transition for me. I have encountered what I believe to be a huge limitation with ASP.NET and I'm wondering how you have handled this problem.

Let's suppose you have a page where the user can do one of two things: add a "Category", or add a "Movie Title" to an already existing category. This is just a make-believe scenario that I'm using to make a point: You have two distinct sections to your page... one section contains a text box and a button (for adding a new category) and another section contains a dropdown menu of categories, a text box, and a different button (for adding a new movie title).

You can have the Click event of the first button ("bAddCategory") handle the addition of the new category, and you can have the Click event of the second button ("bAddMovie") add the new movie title. So far, so good.

My problem is this: what if the user presses the [ENTER] key on his/her keyboard? Doing this does not fire either Click event. I can (in essence) make a "default" button by adding the following to the server-side Load event for the page:

```
Page.RegisterHiddenField("__EVENTTARGET", bAddMovie.ClientID)
```

Doing this will cause the Click event for bAddMovie to fire when the user presses [ENTER]. However... I don't want bAddMovie to be the default button if the user is in the process of typing a new category name (in that case, I obviously would want bAddCategory to be the default).

In traditional ASP, I would simply have two <form>s on the page. One <form> would encapsulate the widgets for entering a category and a second <form> would encapsulate the widgets for entering a movie. Each of the two <form> tags would contain a <input type="hidden"> element containing a name/value pair indicating which form was submitted on postback.

microsoft.public.dotnet.framework.aspnet: [ENTER] key causing major headaches.

In ASP.NET, I see no way to emulate this concept. ASP.NET does not let me have two form tags (with `runat="server"` defined). And, if I use a client-side (traditional) `<form>`, I cannot use any of the ASP.NET web controls (like `<asp:DropDownList>`, etc.). Not being able to use these web controls, to me, defeats the whole purpose of using ASP.NET. If I revert back to a traditional `<form>` tag, I don't get events, viewstates, or any other fancy ASP.NET features.

This concept (of having two forms) is extremely important to me. What do I do!? Any and all help is very much appreciated!!

Regards,

Eric