IdeaSparks ASP.NET CoolControls

Set Focus On Selected Row

vtamas
Today at 8:49 AM

Hi I have a problem with focus on GridView. How can I set the focus on the selected row after postback? Any idea how to solve it? Thanks.

mkamoski
Today at 9:27 AM

> ***vtamas wrote:***
>
> *Hi I have a problem with focus on GridView. How can I set the focus on the selected row after postback? Any idea how to solve it? Thanks.*

Vtam --

Your question concerned me because I need this functionality to work for my current job.

As such, I ran some tests.

Actually, tests indicate that the CoolGrid already does this intrinsically, without having to do anything special.

Please see the test code below to see this work.

As such, it appears that your situation may be different a bit-- can you describe what it is you are doing and what you want to achieve, in more detail, perhaps with a code sample?

This is the code-infront...

```
<%@ page language="C#" autoeventwireup="true" codefile="CoolGridTestPage04.aspx.cs" inherits="CoolGridTestPage04" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ register assembly="IdeaSparx.CoolControls.Web" namespace="IdeaSparx.CoolControls.Web" tagprefix="iscc" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
        <title></title>
        <link rel="Stylesheet" href="Default.css" />
</head>
<body>
        <form id="form1" runat="server">
        <div>
                <p>
                        <a href="Default.aspx">Default.aspx</a>
                </p>
                <p>
                        <asp:label id="MessageLabel" runat="server"></asp:label> 
                </p>
                <p>
                        This page shows a test to demonstrate that the selected-index persists after postback.
                </p>
                <p>
                        Select a row and then click on the PostBack button to see it work.
                </p>
                <p>
                        <asp:button id="PostBackButton" runat="server" text="Fire A PostBack Now" />
                </p>
                <p>
                        <iscc:coolgridview id="MainGridView" runat="server" autogeneratecolumns="false" allowsorting="True" height="500px" width="950px" allowpaging="false" emptydatatext="NULL" cssclass="IsccGr
                                <boundarystyle bordercolor="Transparent" borderwidth="0px" borderstyle="None"></boundarystyle>
                                <alternatingrowstyle cssclass="IsccGridAlternateRowStyle" />
                                <rowstyle cssclass="IsccGridRowStyle" />
                                <selectedrowstyle backcolor="Yellow" />
                                <columns>
                                        <asp:commandfield selecttext="Select" buttontype="Link" showselectbutton="true" />
                                        <asp:boundfield datafield="chargenumber" headertext="Charge Number" readonly="True" sortexpression="chargenumber">
                                                <headerstyle width="100px" />
                                        </asp:boundfield>
                                        <asp:boundfield datafield="divisionname" headertext="Division Name" readonly="True" sortexpression="divisionname">
                                                <headerstyle width="100px" />
                                        </asp:boundfield>
```

```
                                        <asp:boundfield datafield="fullname" headertext="Full Name" readonly="True" sortexpression="fullname">
                                                <headerstyle width="100px" />
                                        </asp:boundfield>
                                        <asp:boundfield datafield="id" headertext="ID" readonly="True" sortexpression="id">
                                                <headerstyle width="100px" />
                                        </asp:boundfield>
                                        <asp:boundfield datafield="organizationname" headertext="Organization Name" readonly="True" sortexpression="organizationname">
                                                <headerstyle width="100px" />
                                        </asp:boundfield>
                                        <asp:boundfield datafield="processreason" headertext="Process Reason" readonly="True" sortexpression="processreason">
                                                <headerstyle width="100px" />
                                        </asp:boundfield>
                                        <asp:boundfield datafield="worksite" headertext="Work Site" readonly="True" sortexpression="worksite">
                                                <headerstyle width="100px" />
                                        </asp:boundfield>
                                        <asp:boundfield datafield="yellowindexcolor" headertext="Yellow Index Color" readonly="True" sortexpression="yellowindexcolor">
                                                <headerstyle width="100px" />
                                        </asp:boundfield>
                                        <asp:boundfield datafield="zebraindexcolor" headertext="Zebra Index Color" readonly="True" sortexpression="zebraindexcolor">
                                                <headerstyle width="100px" />
                                        </asp:boundfield>
                                </columns>
                        </iscc:coolgridview>
                </p>
        </div>
        </form>
</body>
</html>
```

This is the code-behind...

```
using System;
using System.Collections;
using System.Collections.Generic;
using System.Diagnostics;
using System.IO;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Xml;
using System.Xml.Linq;

/// <summary>
/// This page shows a test to demonstrate that the selected-index persists after postback.
/// </summary>
public partial class CoolGridTestPage04 : System.Web.UI.Page
{
        #region MemberEnums

        public enum PersonRecordAttributeName
        {
                isactive
        }

        public enum CollectionName
        {
                personrecords
        }

        public enum TableName
        {
                personrecord
        }

        public enum PersonRecordColumnName
        {
                chargenumber,
                divisionname,
                fullname,
                id,
                organizationname,
                processreason,
                worksite,
                yellowindexcolor,
                zebraindexcolor
        }
```

```
        #endregion //MemberEnums

        #region HandlerMethods

        protected void Page_Load(object sender, EventArgs e)
    {
                try
                {
                        if ((!this.IsPostBack) && ((this.MainGridView == null) ||
                                (this.MainGridView.Rows == null) ||
                                (this.MainGridView.Rows.Count <= 0)))
                        {
                                string myFilePath = Server.MapPath("~");
                                const string myFileName = "TestData01.xml";
                                string myFullPath = Path.Combine(myFilePath, myFileName);
                                Debug.WriteLine("myFullPath='" + myFullPath + "'");
                                XDocument xmlDoc = XDocument.Load(myFullPath);

                                var q = from c in xmlDoc.Descendants(TableName.personrecord.ToString())
                                                where c.Attribute(PersonRecordAttributeName.isactive.ToString()).Value == bool.TrueString.ToLower()
                                                select new
                                                {
                                                        chargenumber = c.Element(PersonRecordColumnName.chargenumber.ToString()).Value,
                                                        divisionname = c.Element(PersonRecordColumnName.divisionname.ToString()).Value,
                                                        fullname = c.Element(PersonRecordColumnName.fullname.ToString()).Value,
                                                        id = c.Element(PersonRecordColumnName.id.ToString()).Value,
                                                        organizationname = c.Element(PersonRecordColumnName.organizationname.ToString()).Value,
                                                        processreason = c.Element(PersonRecordColumnName.processreason.ToString()).Value,
                                                        worksite = c.Element(PersonRecordColumnName.worksite.ToString()).Value,
                                                        yellowindexcolor = c.Element(PersonRecordColumnName.yellowindexcolor.ToString()).Value,
                                                        zebraindexcolor = c.Element(PersonRecordColumnName.zebraindexcolor.ToString()).Value
                                                };

                                this.MainGridView.DataSource = q;
                                this.MainGridView.DataBind();
                        }
                }
                catch (Exception ex)
                {
                        this.MessageLabel.Text = ex.ToString();
                }
        }

        protected void MainGridView_Sorting(object sender, GridViewSortEventArgs e)
        {
                try
                {
                        this.MessageLabel.Text = DateTime.Now.ToString("yyyy-MM-dd hh:mm:ss.ffffff tt zzz") + ": " + "Sorting is not implemented.";
                }
                catch (Exception ex)
                {
                        this.MessageLabel.Text = ex.ToString();
                }

        }

        #endregion //HandlerMethods
}
```

HTH.

Thank you.

-- Mark Kamoski

Reply

Send me an email when someone posts a reply (change email address)
(you are already subscribed to all threads in all discussions change)

© 2006-2010 Microsoft | Get Help | Privacy Statement | Terms of Use | Code of Conduct | Advertise With Us