

# Inline Editing in ASP.NET Example Documentation

This page provides information on how to instantiate EditLive! [Inline Editing](#) within a web page using Active Server Pages in the .NET environment. This is intended as a basic sample only.

## Getting Started

### System Requirements

- Microsoft IIS 6.0 Web Server with ASP.NET 3.0 installed
- The EditLive! ASP.NET Server Control

### Required Skills

The following skills are required prior to working with this sample:

- Developing Web-based forms in HTML
- Developing ASP.NET Web Forms

### Source Code

This documentation includes a copy of the full source code needed for this example. You will find the completed source code in a file called *InlineEditingExample.aspx* in the directory *EDITLIVE\_INSTALL/webfolder/aspnet/inlineEditing*, where *EDITLIVE\_INSTALL* is the location where EditLive! is installed. In order to run this example add the *InlineEditingExample.aspx* file to an ASP.NET Web project which includes a reference to the *EditLiveJavaControl.dll* file.

## Instantiating EditLive! in InlineEditingExample.aspx

For more information on ASP.NET Web Forms, please consult the Microsoft .NET SDK.

1. Perform the same basic steps as seen in the [Basic ASP.NET Example](#) to reference the EditLive! ASP.NET Control and create an instance of the editor.

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="InlineEditingExample.aspx.cs" Inherits="
inlineEditing_InlineEditingExample" ValidateRequest="false" %>
<%@ Register assembly="EditLiveJavaControl" namespace="EditLiveJavaControl" tagprefix="elj" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title>Inline Editing Example</title>
</head>
<body>
  <form id="form1" runat="server" method="post">
    <elj:EditLiveJava ID="EditLiveJava1" runat="server"
      DownloadDirectory="../../redistributables/editlivejava" />
  </form>
</body>
</html>
```

2. Add the [Inline Editing](#) attribute to the **EditLiveJava** tag, specifying *true*.

```
...
  <elj:EditLiveJava ID="EditLiveJava1" runat="server" InlineEditing="true"
    DownloadDirectory="../../redistributables/editlivejava" />
...
```

3. Create an **<elj:EditableSection>** tag for each [Inline Editing](#) section, specifying an ID. Populate the Content attribute with the content you'd like loaded in this [Inline Editing](#) section.



Previously, a separate `<div>` tag was required for each **<elj:EditableSection>** tag. The separate `<div>` is no longer needed.

```

...

    <elj:editlivejava id="EditLiveJava1" runat="server" InlineEditing="True"
    DownloadDirectory="../../redistributables/editlivejava /><br />

    <elj:EditableSection ID="EditableSection1" Content="&lt;p&gt;content 1&lt;/p&gt;"
    runat="server" Height="200px" /><br />
    <elj:EditableSection ID="EditableSection2" Content="&lt;p&gt;content 2&lt;/p&gt;"
    runat="server" Height="200px" /><br />
...

```

## Retrieving Content from EditLive! in InlineEditingExample.aspx

Extracting content from EditLive! [Inline Editing](#) sections is the same as for standard ASP.NET controls. You simply retrieve the value of the Content attribute on the EditableSection control.

In our example, the event-handler for a button populates two server-side <div> tags with the content of our two [Inline Editing](#) sections.

### aspx page:

```

...
<form id="form1" runat="server" method="post">
    <elj:EditLiveJava ID="EditLiveJava1" runat="server" InlineEditing="true"
    DownloadDirectory="../../redistributables/editlivejava" />
    <elj:EditableSection ID="EditableSection1" Content="&lt;p&gt;content 1&lt;/p&gt;"
    runat="server" Height="200px" /><br />
    <elj:EditableSection ID="EditableSection2" Content="&lt;p&gt;content 2&lt;/p&gt;"
    runat="server" Height="200px" /><br />

    <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="Button" /><br />

    <asp:TextBox ID="TextBox1" runat="server" Height="91px" TextMode="MultiLine"
    Width="256px" /><br />
    <asp:TextBox ID="TextBox2" runat="server" Height="88px" TextMode="MultiLine"
    Width="255px" />

</form>
...

```

### c# codebehind:

```

using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;

public partial class inlineEditing_InlineEditingExample : System.Web.UI.Page {
    protected void Button1_Click(object sender, EventArgs e) {
        TextBox1.Text = EditableSection1.Content;
        TextBox2.Text = EditableSection2.Content;
    }
}

```



By default ASP.NET forms cannot post values containing HTML. In order to successfully post EditLive!'s content, you will need to ensure the page directive contains the **validateRequest="false"** parameter.

In ASP.NET 4.0, you also need to add the following to your web.config, in the <system.web> section:

```
<httpRuntime requestValidationMode="2.0" />
```

