This page is specific to

Microsoft Visual Studio 2010/.NET Framework 4

**How to: Implement Callbacks in ASP.NET Web Pages**

In a client callback, a client script function sends a request to the ASP.NET Web page, which then runs an abbreviated version of its normal life cycle to process the callback. To ensure that callback events originate from the expected user interface (UI), you can validate callbacks. In callback validation, you register an event for validation during the Web page rendering and then validate the event during the callback. For an overview of callbacks, see [Implementing Client Callbacks Programmatically Without Postbacks in ASP.NET Web Pages](http://msdn.microsoft.com/en-us/library/ms178208.aspx).

**To implement the ICallBackEventHandler interface**

1. For a single-file page or user control, implement the [ICallbackEventHandler](http://msdn.microsoft.com/en-us/library/system.web.ui.icallbackeventhandler.aspx) interface using an [@ Implements](http://msdn.microsoft.com/en-us/library/cbsf6k72.aspx) directive in the page, as shown in the following example.

Visual Basic

<%@ Page Language="VB" %>

<%@ Implements Interface="System.Web.UI.ICallbackEventHandler" %>

C#

<%@ Page Language="C#" %>

<%@ Implements Interface="System.Web.UI.ICallbackEventHandler" %>

|  |
| --- |
| **Description: NoteNote** |
| If you are using a code-behind page model, implement the [ICallbackEventHandler](http://msdn.microsoft.com/en-us/library/system.web.ui.icallbackeventhandler.aspx) interface for your partial class. |

1. Implement the [RaiseCallbackEvent](http://msdn.microsoft.com/en-us/library/system.web.ui.icallbackeventhandler.raisecallbackevent.aspx) method of the [ICallbackEventHandler](http://msdn.microsoft.com/en-us/library/system.web.ui.icallbackeventhandler.aspx) interface. The [RaiseCallbackEvent](http://msdn.microsoft.com/en-us/library/system.web.ui.icallbackeventhandler.raisecallbackevent.aspx) method takes a single argument that represents the event arguments, as shown in the following example.

Visual Basic

Public Sub RaiseCallbackEvent(ByVal eventArgument As String) \_

Implements System.Web.UI.ICallbackEventHandler.RaiseCallbackEvent

End Sub

C#

public void RaiseCallbackEvent(String eventArgument)

{

}

1. Implement the [GetCallbackResult](http://msdn.microsoft.com/en-us/library/system.web.ui.icallbackeventhandler.getcallbackresult.aspx) method of the [ICallbackEventHandler](http://msdn.microsoft.com/en-us/library/system.web.ui.icallbackeventhandler.aspx) interface. The [GetCallbackResult](http://msdn.microsoft.com/en-us/library/system.web.ui.icallbackeventhandler.getcallbackresult.aspx) method takes no arguments and returns a string that represents the result of the callback. In the following example, a string called returnValue is returned.

Visual Basic

Public Function GetCallbackResult() \_

As String Implements \_

System.Web.UI.ICallbackEventHandler.GetCallbackResult

Return returnValue

End Function

C#

public String GetCallbackResult()

{

return returnValue;

}

**To register the callback for event validation**

* Override the [Render](http://msdn.microsoft.com/en-us/library/20b8wk8d.aspx) method of the [Page](http://msdn.microsoft.com/en-us/library/system.web.ui.page.aspx) class and use the [RegisterForEventValidation](http://msdn.microsoft.com/en-us/library/system.web.ui.clientscriptmanager.registerforeventvalidation.aspx) method of the [ClientScriptManager](http://msdn.microsoft.com/en-us/library/system.web.ui.clientscriptmanager.aspx) class to register an event for validation. You can get a reference to the [ClientScriptManager](http://msdn.microsoft.com/en-us/library/system.web.ui.clientscriptmanager.aspx) class by using the [ClientScript](http://msdn.microsoft.com/en-us/library/system.web.ui.page.clientscript.aspx) property of the [Page](http://msdn.microsoft.com/en-us/library/system.web.ui.page.aspx) class. In the following example, a callback named Callback1 is registered for event validation.

Visual Basic

Protected Overrides Sub Render(ByVal writer As System.Web.UI.HtmlTextWriter)

Page.ClientScript.RegisterForEventValidation("ClientCallback1")

MyBase.Render(writer)

End Sub

C#

protected override void Render(HtmlTextWriter writer)

{

Page.ClientScript.RegisterForEventValidation("ClientCallback1");

base.Render(writer);

}

**To validate the callback and return the callback result**

* In the [RaiseCallbackEvent](http://msdn.microsoft.com/en-us/library/system.web.ui.icallbackeventhandler.raisecallbackevent.aspx) method, use the [ValidateEvent](http://msdn.microsoft.com/en-us/library/system.web.ui.clientscriptmanager.validateevent.aspx) method of the [ClientScriptManager](http://msdn.microsoft.com/en-us/library/system.web.ui.clientscriptmanager.aspx) class to validate the event. Use the same method signature as the one used when registering the event for validation. In the following example, the method signature that takes one argument is used.

Visual Basic

Public Sub RaiseCallbackEvent(ByVal eventArgument As String) \_

Implements System.Web.UI.ICallbackEventHandler.RaiseCallbackEvent

Try

Page.ClientScript.ValidateEvent("ClientCallback1")

' Callback logic goes here.

returnValue = "callback result"

Catch

' Failed callback validation logic.

End Try

End Sub

C#

public void RaiseCallbackEvent(String eventArgument)

{

try

{

Page.ClientScript.ValidateEvent("ClientCallback1");

// Callback logic goes here.

returnValue = "callback result";

}

catch

{

// Failed callback validation logic.

}

}

If validation passes, your code should proceed with the callback event logic. After the [RaiseCallbackEvent](http://msdn.microsoft.com/en-us/library/system.web.ui.icallbackeventhandler.raisecallbackevent.aspx) method has completed, the [GetCallbackResult](http://msdn.microsoft.com/en-us/library/system.web.ui.icallbackeventhandler.getcallbackresult.aspx) method is invoked to return the callback result as a string to a client script function.

For more information about creating and implementing the client script functions to support client callbacks, see [Implementing Client Callbacks Programmatically Without Postbacks in ASP.NET Web Pages](http://msdn.microsoft.com/en-us/library/ms178208.aspx) and [Client Callback with Validation Implementation Example](http://msdn.microsoft.com/en-us/library/ms366515.aspx).