



PIEAS

Pakistan Institute of Engineering and Applied Sciences

Course: Internet Application Development

Lab: # 02

Submitted to: Sir Irfan Hameed

Submitted by: Muhammad Hamza Azhar

Date of Submission: 28th Feb, 2024

**Department of Computer & Information Sciences,
PIEAS, Islamabad, Pakistan**

Develop a page to demonstrate various events which are raised and handled during ASP.NET Page processing once the page object has been created? Develop a story about it?

```
<%@ Page Language="VB" AutoEventWireup="false" CodeFile="Default.aspx.vb"
Inherits="_Default" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>ASP.NET Page Lifecycle Demo</title>
    <style>
        body {
            font-family: 'Arial', sans-serif;
            background-color: #f4f4f4;
            margin: 0;
        }

        form {
            text-align: center;
            margin-top: 50px;
        }

        #btnTrigger {
            padding: 10px;
            font-size: 16px;
            background-color: #4CAF50;
            color: #ffffff;
            border: none;
            cursor: pointer;
            border-radius: 5px;
        }

        #lblOutput {
            display: block;
            max-width:400px;
            position:relative;
            left:33%;
            margin-top: 20px;
            font-size: 18px;
            color: #333;
            border: 1px solid #000;
            border-radius:10px;
            padding: 10px;
        }
    </style>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Button ID="btnTrigger" runat="server" Text="Trigger Postback"
/>
            <br />
            <asp:Label ID="lblOutput" runat="server" Text=""></asp:Label>
        </div>
    </form>
</body>
```

```
</html>
```

```
Partial Class _Default
    Inherits System.Web.UI.Page
    Protected Sub Page_Init(ByVal sender As Object, ByVal e As EventArgs)
Handles Me.Init
        lblOutput.Text &= "Init event handler called<br />"
    End Sub

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As EventArgs)
Handles Me.Load
        lblOutput.Text &= "Load event handler called<br />"
    End Sub

    Protected Sub Page_PreRender(ByVal sender As Object, ByVal e As EventArgs)
Handles Me.PreRender
        lblOutput.Text &= "PreRender event handler called<br />"
    End Sub

    Protected Sub btnTrigger_Click(ByVal sender As Object, ByVal e As
EventArgs) Handles btnTrigger.Click
        lblOutput.Text &= "Button Click event handler called<br />"
    End Sub

    Protected Sub Page_Unload(ByVal sender As Object, ByVal e As EventArgs)
Handles Me.Unload
        lblOutput.Text &= "Unload event handler called<br />"
    End Sub
End Class
```

Story:

In the realm of developers, Hamza, a skilled coder, was tasked to understand ASP.NET Page Life Cycle and its various event handlers. He initiated the process by creating a Default.aspx file and added controls, such as Page_Init, Page_Load, Page_PreLoad, Page_Unload, and a custom event handler, btnTrigger_Click.

Processing Sequence:

- First the browser is launched.
- The client requests the server, and the server responds to the client.
- Page_init processes initially, followed by page_load. After that, page_prerender takes care of rendering the HTML, which is then sent to the client. Finally, page_unload is called, destroying the page object.

Trigger Postback

Init event handler called
Load event handler called
PreRender event handler called

- But when the button is clicked, there's a twist. While `page_init` is called again, the text won't be displayed due to view-state. It's worth noting that view-state is generated on the server side and sent to the client along with the response. The client doesn't directly manipulate the view-state.
- Continuing with the button click, `page_load` is called, and then the `btnTrigger_Click` event handler fires. Next up is the `page_prerender` event, resulting in the response being sent to the client. Finally, `page_unload` comes into play, ensuring the destruction of the created page object.

Note: Because of compound assignment the text will not re-write but it will append.

Trigger Postback

Init event handler called
Load event handler called
PreRender event handler called
Load event handler called
Button Click event handler called
PreRender event handler called

Hamza, through his testing, observed how these events unfolded during the ASP.NET Page processing cycle. He witnessed the initialization of the page, loading and displaying of data, handling of user input, and the subsequent cleanup of resources.