

Tipo : Guía de laboratorio
Capítulo : Controles de datos en WebForms
Duración : 45 minutos

I. OBJETIVO

Diseñar una aplicación Web con controles de datos

II. REQUISITOS

Los siguientes elementos de software son necesarios para la realización del laboratorio:

- Windows 8.1
- SQL Server 2016
- Visual Studio 2017

III. DESCRIPCIÓN

• Ejercicio 1

Usar el control GridView y SqlDataSource para crear el mantenimiento de la tabla Artist.

Actividades

1. Abrir la solución del capítulo anterior.
2. En el fichero **ListArtist.aspx**, insertar las siguientes líneas de código:

```
<%@ Page Title="" Language="C#"
MasterPageFile="~/Site/MantainTemplate.master" AutoEventWireup="true"
CodeBehind="ListArtist.aspx.cs"
Inherits="WebForms.Site.ArtistWeb.ListArtist" %>

<asp:Content ID="ButtonContent" ContentPlaceHolderID="ButtonContent"
runat="server">
    <a class="btn btn-info" href="CreateArtist.aspx">
        <span class="glyphicon glyphicon-plus"></span> Crear Artist
    </a>
</asp:Content>
<asp:Content ID="ListContent" ContentPlaceHolderID="ArtistContent"
runat="server">
    <div class="row">
        <h3>Artist List</h3>
    </div>
    <asp:SqlDataSource ID="SqlDataSource"
        runat="server"
        ConnectionString="<%$ ConnectionStrings:ChinookConnection %>"
        SelectCommand="GetListOfArtist"
        SelectCommandType="StoredProcedure"
        DeleteCommand="DeleteArtist"
        DeleteCommandType="StoredProcedure">
        <DeleteParameters>
            <asp:Parameter Name="artistId" Type="Int32" />
        </DeleteParameters>
    </asp:SqlDataSource>
</asp:Content>
```

```

        </DeleteParameters>
    </asp:SqlDataSource>

    <div class="row">
        <br />
        <asp:GridView ID="artistGridView"
            runat="server"
            AllowPaging="True"
            AllowSorting="True"
            AutoGenerateColumns="True"
            DataSourceID="SqlDataSource"
            AutoGenerateDeleteButton="True"
            AutoGenerateEditButton="True">
        </asp:GridView>
        <br />
    </div>

</asp:Content>

```

3. En el fichero **CreateArtist.aspx**, agregar el siguiente código:

```

<asp:Content ID="ButtonContent" ContentPlaceHolderID="ButtonContent"
runat="server">
    <a class="btn btn-info" href="ListArtist.aspx">
        <span class="glyphicon glyphicon-hand-left"></span> Return
    </a>
</asp:Content>
<asp:Content ID="ArtistContent" ContentPlaceHolderID="ArtistContent"
runat="server">
    <div class="row">
        <div>
            <asp:Label ID="Label1" runat="server" Text="Artist Name:"
        />
            <asp:TextBox ID="txtName" runat="server"></asp:TextBox>
        </div>
    </div>
    <div class="row">
        <div>
            <asp:Button CssClass="btn btn-submit" runat="server"
ID="btnCreate" Text="Create" OnClick="btnCreate_Click" />
        </div>
    </div>
</asp:Content>

```

4. En el code behind del fichero **CreateArtist.aspx**, archivo **CreateArtist.aspx.cs** insertar las siguientes líneas de código:

```

using DataAccess;
using Models;
using System;

namespace WebForms.Site.ArtistWeb
{
    public partial class CreateArtist : System.Web.UI.Page
    {
        private UnitOfWork _unit;
        public CreateArtist()
        {
            _unit = new UnitOfWork(new ChinookContext());
        }
        protected void Page_Load(object sender, EventArgs e)
        {

```

```

    }

    protected void btnCreate_Click(object sender, EventArgs e)
    {
        var artist = new Artist { Name=txtName.Text };
        _unit.Artists.Add(artist);
        if(_unit.Complete()>0)
        {
            Response.Redirect("ListArtist.aspx");
        }
    }
}
}
}

```

5. Ejecutar la aplicación, validar que la edición, eliminación y creación sea funcional.

IV. EVALUACIÓN

1. ¿Qué elementos HTML generan los controles GridView y el Repeater al ser visualizados en un browser? (Revisar HTML generado con Firebug o herramientas de Google Chrome).