

Ejercicio: Implementar un aplicativo que permita listar las fichas de devolución por contratista.

- Crear un procedimiento almacenado para listar
- Configurar la cadena de conexión en el archivo app.config
- Crear la clase Conexión y un método para asociarse a la cadena de conexión configurada en app.config
- Crear el SP "SP_CONTRATISTA", que liste los contratistas y Otro "SP_LISTAFICHADEVOLUCIONXCONTRATISTA"

Procedimientos almacenados

```
-- Procedimiento para lista de contratistas
IF OBJECT_ID('SP_CONTRATISTA') IS NOT NULL
    DROP PROC SP_CONTRATISTA
GO

CREATE PROC SP_CONTRATISTA
AS
    SELECT C.IDE_CON AS CODIGO,
           C.NOM_CON + SPACE(1) + C.PAT_CON + SPACE(1) + C.MAT_CON AS
CONTRATISTA
    FROM CONTRATISTA C
GO

-- Procedimiento para listar las fichas de devolución por contratista (código)
IF OBJECT_ID('SP_LISTAFICHADEVOLUCIONXCONTRATISTA') IS NOT NULL
    DROP PROC SP_LISTAFICHADEVOLUCIONXCONTRATISTA
GO

CREATE PROC SP_LISTAFICHADEVOLUCIONXCONTRATISTA(@CON CHAR(6))
AS
    SELECT FD.COD_FIC AS CODIGO,
           C.NOM_CON + SPACE(1) + C.PAT_CON + SPACE(1) + C.MAT_CON AS
CONTRATISTA,
           CL.NOM_CLI + SPACE(1) + CL.PAT_CLI + SPACE(1) + CL.MAT_CLI AS
CLIENTE,
           E.DESC_EQU AS EQUIPO,
           FD.FDE_FIC AS FECHA
    FROM FICHA_DEVOLUCION FD
    JOIN CLIENTE CL ON FD.IDE_CLI = CL.IDE_CLI
    JOIN CONTRATISTA C ON FD.IDE_CON = C.IDE_CON
    JOIN EQUIPO E ON FD.IDE_EQU = E.IDE_EQU
    WHERE FD.IDE_CON = @CON
GO
```

Agregar la referencia "System.configuration"

Configurar la cadena de conexión en el archivo "App.config"

```
<configuration>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5.1" />
  </startup>

  <!-- Cadena de conexión-->
  <connectionStrings>
    <add name="cn" connectionString="server=.;
                                     database=CONTRATO;
                                     integrated security=SSPI"/>
  </connectionStrings>
</configuration>
```

Crear la clase "Conexion"

```
namespace pJFichaDevolucionXContratista
{
    2 references
    class Conexion
    {
        // Metodo que devuelve la conexión
        2 references
        public SqlConnection getConexion()
        {
            SqlConnection sqlConnection = new SqlConnection(
                ConfigurationManager.ConnectionStrings["cn"].ConnectionString);
            return sqlConnection;
        }
    }
}
```

Crear la clase "LogicaNegocio"

```
2 references
class LogicaNegocio
{
    // Variables globales
    private Conexion conexion = new Conexion();
    private SqlConnection sqlConnection;

    // Método para listar los contratista (para el combo)
    1 reference
    public DataTable contratistasCbo()
    {
        sqlConnection = conexion.getConexion(); // Conexión a la BD
        // Para ejecutar el query o SP
        string sql = "SELECT C.IDE_CON AS CODIGO,"
        + "C.NOM_CON + SPACE(1) + C.PAT_CON + SPACE(1) + C.MAT_CON AS CONTRATISTA FROM CONTRATISTA C";
        SqlDataAdapter sqlDataAdapter = new SqlDataAdapter(
            sql, sqlConnection);
        // Llenar al DataTable
        DataTable dataTable = new DataTable();
        sqlDataAdapter.Fill(dataTable); // Llenar al dataTable
        return dataTable; // Retorna la data
    }
}
```

```

// Método para listar las fichas de devolución X contratista
1 reference
public DataTable listarFichasXContratista(string contratista)
{
    sqlConnection = conexion.getConexion(); // Conexión a la BD
    sqlConnection.Open(); // Abrir la conexión
    SqlDataAdapter sqlDataAdapter = new SqlDataAdapter(
        "SP_LISTAFICHADEVOLUCIONXCONTRATISTA", sqlConnection);
    // El SP requiere de un parámetro, por esta razón, cambia un poco las líneas
    sqlDataAdapter.SelectCommand.CommandType = CommandType.StoredProcedure; //Tipo
    // Agregando parámetro
    sqlDataAdapter.SelectCommand.Parameters.Add("CON", SqlDbType.Char).Value = contratista;
    DataTable dataTable = new DataTable();
    sqlDataAdapter.Fill(dataTable); // Llenar al dataTable
    return dataTable; // Retorna la data
}
}

```

Formulario

```

namespace pjFichaDevolucionXContratista
{
    3 references
    public partial class Form1 : Form
    {
        // Objeto Logica
        private LogicaNegocio logicaNegocio = new LogicaNegocio();
        1 reference
        public Form1()
        {
            InitializeComponent();
        }

        // Método para llenar el combo de contratistas
        1 reference
        private void contratistasCbo()
        {
            // Mismo nombre con que está en query o el SP
            cboContratista.DataSource = logicaNegocio.contratistasCbo();
            cboContratista.DisplayMember = "CONTRATISTA";
            cboContratista.ValueMember = "CODIGO";
        }
    }
}

```

```
1 reference
private void Form1_Load(object sender, EventArgs e)
{
    contratistasCbo();
}

// Métoodo para llenar la tabla según contratista
1 reference
private void cboContratista_SelectionChangeCommitted(object sender, EventArgs e)
{
    // Captando contratista
    string contratista = cboContratista.SelectedValue.ToString();
    dgFichaDevolucion.DataSource = logicaNegocio.listarFichasXContratista(contratista);
}
}
```