Ejercicio de Integrador de XML

Integrantes del grupo:

* Goncalves, Lucas
* Jorreto, Mariano
* López, Ariel
* Schiter, Tomás
* Zurriable, Jorge

Todos los puntos del enunciado fueron resueltos dentro de la misma página “Default.aspx”. Para una mayor claridad de la documentación, las partes del código involucradas en la resolución de cada uno de los puntos del enunciado tendrá su correspondiente comentario indicando qué punto es el que se está resolviendo. Los puntos resueltos del enunciado son los siguientes.

* Lectura de datos con XmlTextReader.
* Escritura de datos con XmlTextWriter.
* Búsqueda utilizando XPath.
* Búsqueda utilizando XSLT.

Si bien somos conscientes de que el último punto del enunciado requería hacer una búsqueda utilizando XPath **o** XSLT, decidimos hacerlo de las dos formas.

**Libros.xml**

<?xml version="1.0"?>

<Libros>

<Libro>

<NombreLibro>Origen</NombreLibro>

<Autor>Dan Brown</Autor>

<Editorial>Planeta</Editorial>

<Precio>1490</Precio>

</Libro>

<Libro>

<NombreLibro>El retorno de los dragones</NombreLibro>

<Autor>Margaret Weis - Tracy Hickman</Autor>

<Editorial>Timun Mas</Editorial>

<Precio>250</Precio>

</Libro>

<Libro>

<NombreLibro>La herida</NombreLibro>

<Autor>Jorge Fernandez Diaz</Autor>

<Editorial>Planeta</Editorial>

<Precio>600</Precio>

</Libro>

<Libro>

<NombreLibro>Ajuste de cuentas</NombreLibro>

<Autor>John Grisham</Autor>

<Editorial>Penguin Random House</Editorial>

<Precio>1424</Precio>

</Libro>

</Libros>

**Busqueda.xslt**

<?xml version="1.0" encoding="utf-8"?>

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"

xmlns:msxsl="urn:schemas-microsoft-com:xslt" exclude-result-prefixes="msxsl"

>

<xsl:param name="ParamEditorial"></xsl:param>

<xsl:template match="/">

<table border="2" cellpadding="5">

<thead>

<th>Nombre</th>

<th>Autor</th>

<th>Editorial</th>

<th>Precio</th>

</thead>

<tbody>

<xsl:for-each select="Libros/Libro[Editorial=$ParamEditorial]">

<tr align="center">

<td>

<xsl:value-of select="NombreLibro"/>

</td>

<td>

<xsl:value-of select="Autor"/>

</td>

<td>

<xsl:value-of select="Editorial"/>

</td>

<td>

<xsl:value-of select="Precio"/>

</td>

</tr>

</xsl:for-each>

</tbody>

</table>

</xsl:template>

</xsl:stylesheet>

**Default.aspx**

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="EjercicioIntegradorXml.Default" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>

<title></title>

</head>

<body>

<form id="form1" runat="server">

<div>

<h3>Lectura con XmlTextReader</h3>

<asp:Button ID="btnLeerConXmlTextReader" runat="server" Text="Leer" OnClick="BtnLeerConXmlTextReader\_Click" />

<br />

<div>

<asp:Label ID="lblDatos" runat="server"></asp:Label>

</div>

<hr />

<h3>Escritura con XmlTextWriter</h3>

Nombre: <asp:TextBox ID="txtNombreLibro1" runat="server"></asp:TextBox>

Autor: <asp:TextBox ID="txtAutorLibro1" runat="server"></asp:TextBox>

Editorial: <asp:TextBox ID="txtEditorialLibro1" runat="server"></asp:TextBox>

Precio: <asp:TextBox ID="txtPrecioLibro1" runat="server"></asp:TextBox>

<br />

<asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server" ErrorMessage="El nombre del libro es requerido. " ControlToValidate="txtNombreLibro1" ValidationGroup="XmlTextWriter"></asp:RequiredFieldValidator>

<asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server" ErrorMessage="El autor del libro es requerido. " ControlToValidate="txtAutorLibro1" ValidationGroup="XmlTextWriter"></asp:RequiredFieldValidator>

<asp:RequiredFieldValidator ID="RequiredFieldValidator3" runat="server" ErrorMessage="La editorial del libro es requerida. " ControlToValidate="txtEditorialLibro1" ValidationGroup="XmlTextWriter"></asp:RequiredFieldValidator>

<asp:RequiredFieldValidator ID="RequiredFieldValidator4" runat="server" ErrorMessage="El precio del libro es requerido. " ControlToValidate="txtPrecioLibro1" ValidationGroup="XmlTextWriter"></asp:RequiredFieldValidator>

<br />

Nombre: <asp:TextBox ID="txtNombreLibro2" runat="server"></asp:TextBox>

Autor: <asp:TextBox ID="txtAutorLibro2" runat="server"></asp:TextBox>

Editorial: <asp:TextBox ID="txtEditorialLibro2" runat="server"></asp:TextBox>

Precio: <asp:TextBox ID="txtPrecioLibro2" runat="server"></asp:TextBox>

<br />

<asp:RequiredFieldValidator ID="RequiredFieldValidator5" runat="server" ErrorMessage="El nombre del libro es requerido. " ControlToValidate="txtNombreLibro2" ValidationGroup="XmlTextWriter"></asp:RequiredFieldValidator>

<asp:RequiredFieldValidator ID="RequiredFieldValidator6" runat="server" ErrorMessage="El autor del libro es requerido. " ControlToValidate="txtAutorLibro2" ValidationGroup="XmlTextWriter"></asp:RequiredFieldValidator>

<asp:RequiredFieldValidator ID="RequiredFieldValidator7" runat="server" ErrorMessage="La editorial del libro es requerida. " ControlToValidate="txtEditorialLibro2" ValidationGroup="XmlTextWriter"></asp:RequiredFieldValidator>

<asp:RequiredFieldValidator ID="RequiredFieldValidator8" runat="server" ErrorMessage="El precio del libro es requerido. " ControlToValidate="txtPrecioLibro2" ValidationGroup="XmlTextWriter"></asp:RequiredFieldValidator>

<br />

Nombre: <asp:TextBox ID="txtNombreLibro3" runat="server"></asp:TextBox>

Autor: <asp:TextBox ID="txtAutorLibro3" runat="server"></asp:TextBox>

Editorial: <asp:TextBox ID="txtEditorialLibro3" runat="server"></asp:TextBox>

Precio: <asp:TextBox ID="txtPrecioLibro3" runat="server"></asp:TextBox>

<br />

<asp:RequiredFieldValidator ID="RequiredFieldValidator9" runat="server" ErrorMessage="El nombre del libro es requerido. " ControlToValidate="txtNombreLibro3" ValidationGroup="XmlTextWriter"></asp:RequiredFieldValidator>

<asp:RequiredFieldValidator ID="RequiredFieldValidator10" runat="server" ErrorMessage="El autor del libro es requerido. " ControlToValidate="txtAutorLibro3" ValidationGroup="XmlTextWriter"></asp:RequiredFieldValidator>

<asp:RequiredFieldValidator ID="RequiredFieldValidator11" runat="server" ErrorMessage="La editorial del libro es requerida. " ControlToValidate="txtEditorialLibro3" ValidationGroup="XmlTextWriter"></asp:RequiredFieldValidator>

<asp:RequiredFieldValidator ID="RequiredFieldValidator12" runat="server" ErrorMessage="El precio del libro es requerido. " ControlToValidate="txtPrecioLibro3" ValidationGroup="XmlTextWriter"></asp:RequiredFieldValidator>

<br />

<asp:Button ID="btnEscribirConXmlTextWriter" runat="server" ValidationGroup="XmlTextWriter" Text="Agregar 3 libros" OnClick="BtnEscribirConXmlTextWriter\_Click" />

<br />

<asp:Label ID="lblResultadoEscritura" runat="server"></asp:Label>

<hr />

<h3>Búsqueda por XPath</h3>

Editorial: <asp:TextBox ID="txtBusquedaXpathEditorial" runat="server"></asp:TextBox>

<br /><br />

<asp:Button ID="btnBuscarXPath" runat="server" Text="Buscar con XPath" OnClick="BtnBuscarXPath\_Click" />

<br /><br />

<asp:Label ID="lblBusquedaXpath" runat="server"></asp:Label>

<br />

<asp:ListBox ID="lstBusquedaXpath" runat="server" Width="400px"></asp:ListBox>

<hr />

<h3>Búsqueda por XSLT</h3>

Editorial: <asp:TextBox ID="txtBusquedaXsltEditorial" runat="server"></asp:TextBox>

<br /><br />

<asp:Button ID="btnBuscarXslt" runat="server" Text="Buscar con XSLT" OnClick="BtnBuscarXslt\_Click" />

<br /><br />

<asp:Label ID="lblBusquedaXslt" runat="server"></asp:Label>

</div>

</form>

</body>

</html>

**Default.aspx.cs**

using System;

using System.IO;

using System.Web.UI;

using System.Xml;

using System.Xml.XPath;

using System.Xml.Xsl;

namespace EjercicioIntegradorXml

{

public partial class Default : Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void BtnLeerConXmlTextReader\_Click(object sender, EventArgs e)

{

XmlTextReader miLector = new XmlTextReader(Server.MapPath("Libros.xml"));

bool printText = false;

while (miLector.Read())

{

if (miLector.NodeType == XmlNodeType.Element)

{

switch (miLector.Name)

{

case "Libro":

// Es un libro nuevo, agrego un renglón en blanco

lblDatos.Text += "<br/>";

break;

case "NombreLibro":

lblDatos.Text += "<b>Nombre:</b> ";

// Se pone esta variable en true para que se imprima su valor en la próxima iteración del while

printText = true;

break;

case "Autor":

lblDatos.Text += "<b>Autor:</b> ";

printText = true;

break;

case "Editorial":

lblDatos.Text += "<b>Editorial:</b> ";

printText = true;

break;

case "Precio":

lblDatos.Text += "<b>Precio:</b> $";

printText = true;

break;

}

}

else if (miLector.NodeType == XmlNodeType.Text && printText)

{

printText = false;

lblDatos.Text += miLector.Value + "<br/>";

}

}

miLector.Close();

}

protected void BtnEscribirConXmlTextWriter\_Click(object sender, EventArgs e)

{

XmlTextWriter miEscritor = new XmlTextWriter(Server.MapPath("Libros.xml"), null)

{

Formatting = Formatting.Indented

};

miEscritor.WriteStartDocument();

miEscritor.WriteStartElement("Libros");

AgregarLibro(miEscritor, "Origen", "Dan Brown", "Planeta", "1490");

AgregarLibro(miEscritor, "El retorno de los dragones", "Margaret Weis - Tracy Hickman", "Timun Mas", "250");

AgregarLibro(miEscritor, "La herida", "Jorge Fernandez Diaz", "Planeta", "600");

AgregarLibro(miEscritor, "Ajuste de cuentas", "John Grisham", "Penguin Random House", "1424");

AgregarLibro(miEscritor, txtNombreLibro1.Text, txtAutorLibro1.Text, txtEditorialLibro1.Text, txtPrecioLibro1.Text);

AgregarLibro(miEscritor, txtNombreLibro2.Text, txtAutorLibro2.Text, txtEditorialLibro2.Text, txtPrecioLibro2.Text);

AgregarLibro(miEscritor, txtNombreLibro3.Text, txtAutorLibro3.Text, txtEditorialLibro3.Text, txtPrecioLibro3.Text);

miEscritor.WriteEndDocument();

miEscritor.Flush();

miEscritor.Close();

lblResultadoEscritura.Text = "Listo!";

}

private void AgregarLibro(XmlTextWriter miEscritor, string nombre, string autor, string editorial, string precio)

{

miEscritor.WriteStartElement("Libro");

miEscritor.WriteStartElement("NombreLibro");

miEscritor.WriteString(nombre);

miEscritor.WriteEndElement();

miEscritor.WriteStartElement("Autor");

miEscritor.WriteString(autor);

miEscritor.WriteEndElement();

miEscritor.WriteStartElement("Editorial");

miEscritor.WriteString(editorial);

miEscritor.WriteEndElement();

miEscritor.WriteStartElement("Precio");

miEscritor.WriteString(precio);

miEscritor.WriteEndElement();

miEscritor.WriteEndElement();

}

protected void BtnBuscarXPath\_Click(object sender, EventArgs e)

{

XPathDocument xPathDocum = new XPathDocument(Server.MapPath("Libros.xml"));

XPathNavigator xNavegador = xPathDocum.CreateNavigator();

XPathNodeIterator iterador = xNavegador.Select("Libros/Libro[Editorial='" + txtBusquedaXpathEditorial.Text + "']/NombreLibro");

lblBusquedaXpath.Text = "Nombres de libros de la editorial " + txtBusquedaXpathEditorial.Text;

while (iterador.MoveNext())

{

lstBusquedaXpath.Items.Add(iterador.Current.Value);

}

}

protected void BtnBuscarXslt\_Click(object sender, EventArgs e)

{

XsltArgumentList xsltArguments = new XsltArgumentList();

xsltArguments.AddParam("ParamEditorial", "", txtBusquedaXsltEditorial.Text);

XslCompiledTransform transform = new XslCompiledTransform();

transform.Load(Server.MapPath("Busqueda.xslt"));

StringWriter results = new StringWriter();

transform.Transform(Server.MapPath("Libros.xml"), xsltArguments, results);

string html = results.ToString();

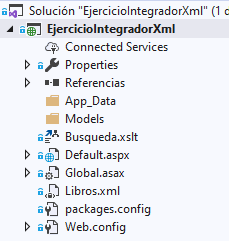
lblBusquedaXslt.Text = html;

}

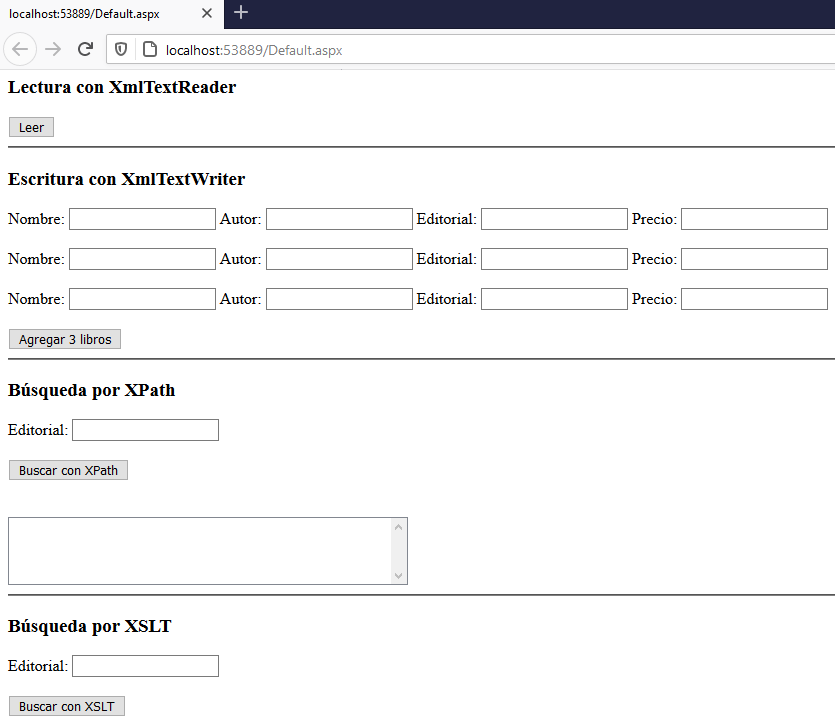
}

}

**Imagen de la solución**

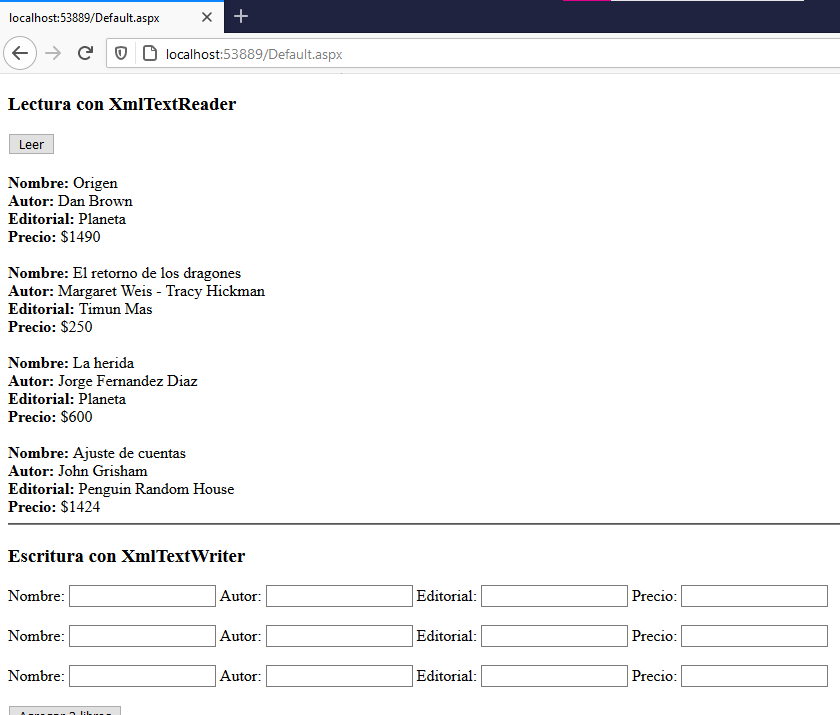


**Ejecución**



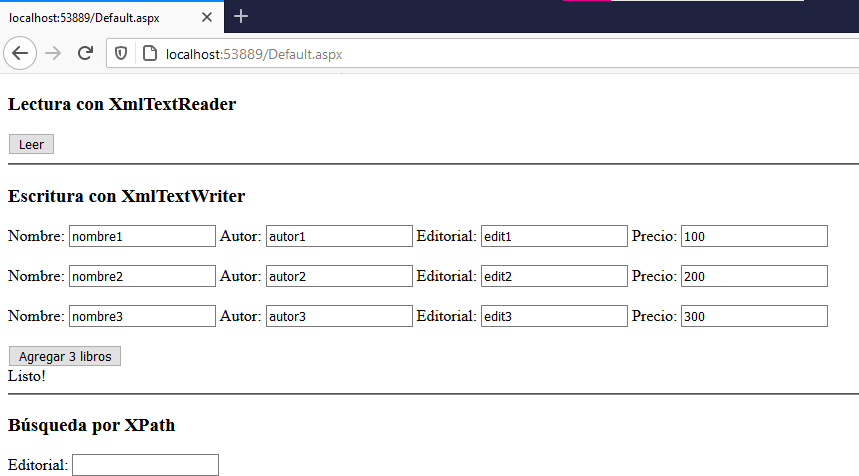
**Lectura con XmlTextReader**

Se presiona el botón “Leer”.

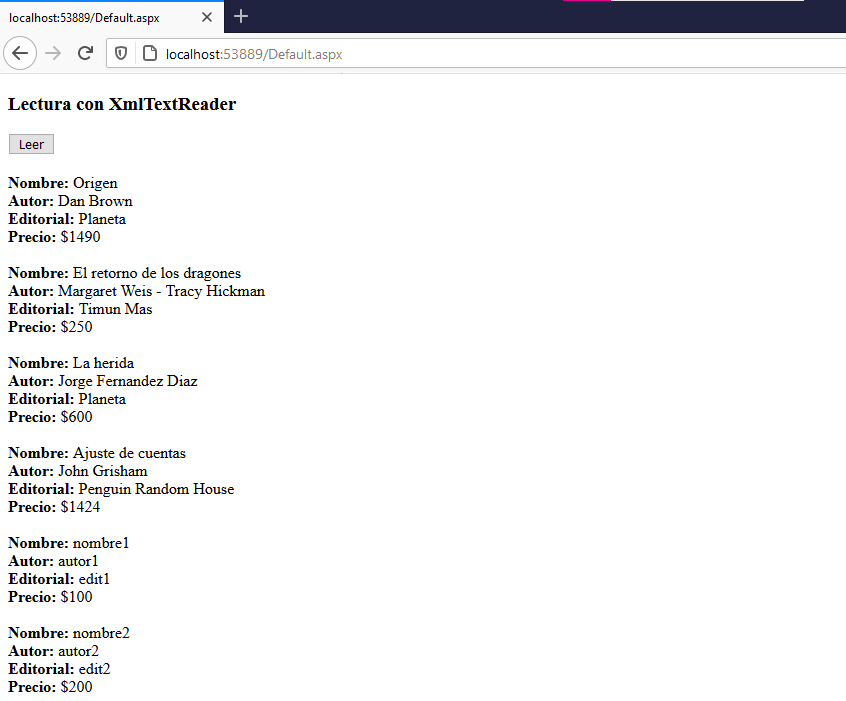


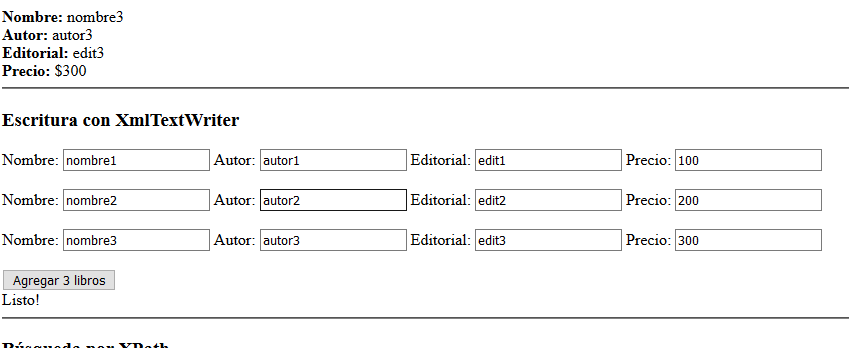
**Escritura con XmlTextWriter**

Se completan los datos y se presiona el botón “Agregar 3 libros”.

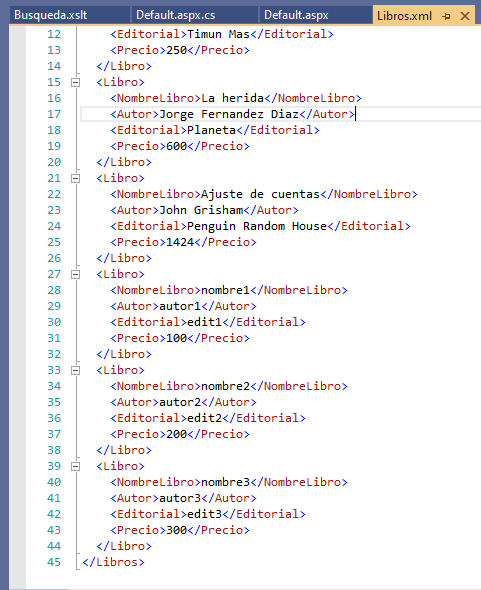


A continuación, se presiona el botón “Leer” (de la sección de la lectura con XmlTextReader) para comprobar que se agregaron los 3 libros al archivo XML.



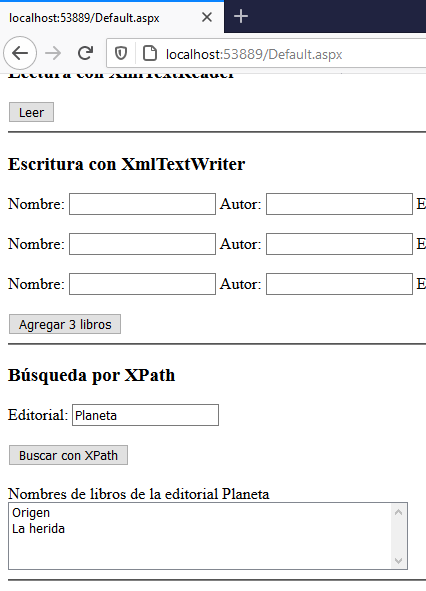


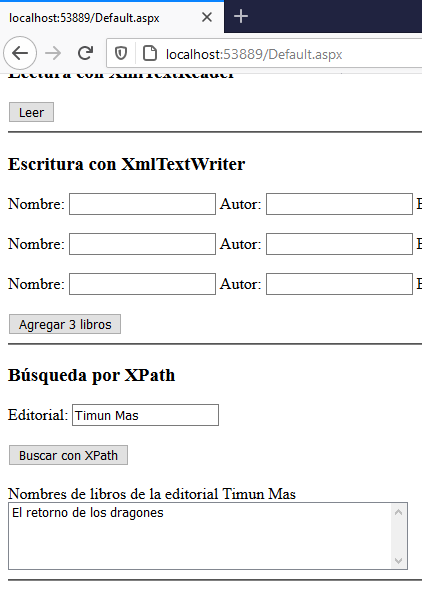
Y se observa el archivo XML en el editor de Visual Studio, donde se verifica que los libros fueron agregados.



**Búsqueda por XPath**

Se completa la Editorial y se presiona el botón “Buscar con XPath”.





**Búsqueda por XSLT**

Se completa la Editorial y se presiona el botón “Buscar con XSLT”.

