How to convert XML to XLS in C# with ByteScout Spreadsheet SDK

Tutorial on how to convert XML to XLS in C#

These sample source codes on this page below are demonstrating how to convert XML to XLS in C#. Want to convert XML to XLS in your C# app? ByteScout Spreadsheet SDK is designed for it. ByteScout Spreadsheet SDK is the library (SDK) that is capable of writing, reading, modifying and calculating Excel and CSV spreadsheets. Most popular formulas can be calculated and reculculated with Excel installed. You may import or export data to and from CSV, XML, JSON as well as to and from databases, arrays.

C# code samples for C# developers help to speed up coding of your application when using ByteScout Spreadsheet SDK. In order to implement the functionality, you should copy and paste this code for C# below into your code editor with your app, compile and run your application. Detailed tutorials and documentation are available along with installed ByteScout Spreadsheet SDK if you'd like to dive deeper into the topic and the details of the API.

You can download free trial version of ByteScout Spreadsheet SDK from our website to see and try many others source code samples for C#.

FOR MORE INFORMATION AND FREE TRIAL:

Download Free Trial SDK (on-premise version)

Read more about ByteScout Spreadsheet SDK

Explore API Documentation

Get Free Training for ByteScout Spreadsheet SDK

Get Free API key for Web API

visit www.ByteScout.com

Source Code Files:

Program.cs

```csharp
using System;
using System.Collections.Generic;
using System.Diagnostics;
using System.Drawing;
using System.Xml;
using Bytescout.Spreadsheet;
using Bytescout.Spreadsheet.Constants;

namespace ConvertXmlToXlsExample
{
        /// <summary>
        /// This example demonstrates how to create Excel spreadsheet from some XML da
        /// Since your XML file has different structure the example just shows techniq
        //  and spreadsheet creation.
        /// </summary>
        class Program
        {
                static void Main()
                {
                        // Load XML document
                        XmlDocument xmlDocument = new XmlDocument();
                        xmlDocument.Load(@"sample.xml");

                        // Read columns information from XML data
                        List<string> columns = new List<string>();
                        XmlNodeList columnNodeList = xmlDocument.SelectNodes("/Report/
                        foreach (XmlNode node in columnNodeList)
                                columns.Add(node.Attributes["Name"].Value);

                        // Read row nodes from XML data
                        XmlNodeList rowNodeList = xmlDocument.SelectNodes("/Report/Rep


                        // Create new spreadsheet
                        Spreadsheet spreadsheet = new Spreadsheet();
                        spreadsheet.RegistrationName = "demo";
                        spreadsheet.RegistrationKey = "demo";
                        // Add worksheet
                        Worksheet worksheet = spreadsheet.Worksheets.Add();

                        // Add column headers
                        for (int c = 0; c < columns.Count; c++)
                        {
                                worksheet[0, c].Value = columns[c];
                                worksheet[0, c].FillPattern = PatternStyle.Solid;
                                worksheet[0, c].FillPatternForeColor = Color.LightGray
                        }

                        int rowIndex = 1;

                        // Add rows
                        foreach (XmlNode rowNode in rowNodeList)
                        {
                                // Get cell values from XML data
                                foreach (XmlNode childNode in rowNode.ChildNodes)
```

```csharp
                    {
                        // Get cell info from XML data
                        int columnIndex = columns.IndexOf(childNode.Nam
                        string cellValue = childNode.InnerText;

                        Cell cell = worksheet[rowIndex, columnIndex];

                        // Set cell text
                        cell.Value = cellValue;
                        // Set cell text alignment
                        cell.AlignmentHorizontal = columnIndex == 0 ? /
                    }

                    // Add the row to the table
                    rowIndex++;
                }

                // Fit columns width to cell data
                for (int c = 0; c < columns.Count; c++)
                        worksheet.Columns[c].AutoFit();

                // Save document to file
                spreadsheet.SaveAsXLS("result.xls");

                // Cleanup
                spreadsheet.Dispose();

                // Open document in Excel
                Process.Start("result.xls");
            }
        }
}
```

VIDEO

https://www.youtube.com/watch?v=nm_7I0PN1TY

ON-PREMISE OFFLINE SDK

60 Day Free Trial or Visit ByteScout Spreadsheet SDK Home Page
Explore ByteScout Spreadsheet SDK Documentation
Explore Samples
Sign Up for ByteScout Spreadsheet SDK Online Training

ON-DEMAND REST WEB API

[Get Your API Key](#)
[Explore Web API Docs](#)
[Explore Web API Samples](#)

[visit www.ByteScout.com](#)

[visit www.PDF.co](#)