

How to convert XML to PDF in C# using ByteScout PDF SDK

How to code in C# to convert XML to PDF with this step-by-step tutorial

The sample shows steps and algorithm of how to convert XML to PDF and how to make it work in your C# application. ByteScout PDF SDK: the pdf library that can create, update and modify PDF files. Supports text with fonts and style selections, layers, form fields, drawing lines and objects, automatic tables, images. Can be used to create and fill pdf forms. It can convert XML to PDF in C#.

C# code samples for C# developers help to speed up coding of your application when using ByteScout PDF SDK. This C# sample code is all you need for your app. Just copy and paste the code, add references (if needs to) and you are all set! Test C# sample code examples whether they respond your needs and requirements for the project.

ByteScout free trial version is available for download from our website. It includes all these programming tutorials along with source code samples.

C# - Program.cs

```
using System;
using System.Collections.Generic;
using System.Diagnostics;
using System.Xml;
using Bytescout.PDF;

namespace ConvertXmlToPdfExample
{
    ///
    /// This example demonstrates how to create table from some XML data.
    /// Since your XML file has different structure the example just shows technique
of XML data reading
    /// and PDF table creation.
    ///

    class Program
    {
        static void Main()
        {
            // Load XML document
            XmlDocument xmlDoc = new XmlDocument();
            xmlDoc.Load(@"sample.xml");

            // Read columns information from XML data
            List columns = new List();
```

```

XmlNodeList columnNodeList =
xmlDocument.SelectNodes("/Report/Columns/Column");
foreach (XmlNode node in columnNodeList)
    columns.Add(node.Attributes["Name"].Value);

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

// Create new PDF document
Document pdfDocument = new Document();
pdfDocument.RegistrationName = "demo";
pdfDocument.RegistrationKey = "demo";
// Add page
Page page = new Page(PaperFormat.A4);
pdfDocument.Pages.Add(page);

DeviceColor lightGrayColor = new ColorGray(200);
DeviceColor whiteColor = new ColorGray(255);

// Create PDF table
Table table = new Table();
table.BackgroundColor = lightGrayColor;

// Add columns
for (int c = 0; c < columns.Count; c++)
{
    TableColumn column = new TableColumn(columns[c], columns[c]);
    // Set column width
    column.Width = (c == 0)? 100 : 60;
    table.Columns.Add(column);
}

// Add rows
foreach (XmlNode rowNode in rowNodeList)
{
    // Create new row and set its background color
    TableRow row = table.NewRow();
    row.BackgroundColor = whiteColor;

    // Get cell values from XML data
    foreach (XmlNode childNode in rowNode.ChildNodes)
    {
        // Get cell info from XML data
        string columnName = childNode.Name;
        int columnIndex = columns.IndexOf(childNode.Name);
        string cellValue = childNode.InnerText;

        // Set cell text
        row[columnName].Text = cellValue;
        // Set cell text alignment
        row[columnName].TextFormat.HorizontalAlign = (columnIndex == 0) ?
HorizontalAlign.Left : HorizontalAlign.Right;
    }

    // Add the row to the table
    table.Rows.Add(row);
}

```

```
// Draw the table on canvas
page.Canvas.DrawTable(table, 20, 20);

// Save document to file
pdfDocument.Save("result.pdf");

// Cleanup
pdfDocument.Dispose();

// Open result document in default associated application (for demo
purpose)
ProcessStartInfo processStartInfo = new ProcessStartInfo("result.pdf");
processStartInfo.UseShellExecute = true;
Process.Start(processStartInfo);
    }
}
```

FOR MORE INFORMATION AND FREE TRIAL:

[Download Free Trial SDK \(on-premise version\)](#)

[Read more about ByteScout PDF SDK](#)

[Explore documentation](#)

[Visit www.ByteScout.com](#)

or

[Get Your Free API Key for www.PDF.co Web API](#)