

How to add page numbers in C# and ByteScout PDF SDK

The tutorial below will demonstrate how to add page numbers in C#

Source code documentation samples provide quick and easy way to add a required functionality into your application. ByteScout PDF SDK: the library for pdf documents generation, modification and updates. Can also generate and fill PDF forms. Provides support for text (fonts, style, size, font family), layers, pdf form fields, vector and raster drawings. It can add page numbers in C#.

You will save a lot of time on writing and testing code as you may just take the C# code from ByteScout PDF SDK for add page numbers below and use it in your application. In your C# project or application you may simply copy & paste the code and then run your app! 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 Bytescout.PDF;
using System;
using System.Diagnostics;
using Font = Bytescout.PDF.Font;
using SolidColorBrush = Bytescout.PDF.SolidBrush;

namespace AddHeaderAndFooter
{
    class Program
    {
        #region Declarations

        private enum PageNoStyle
        {
            TopLeft,
            TopRight,
            BottomLeft,
            BottomRight
        }

        #endregion

        static void Main(string[] args)
        {
            try
            {
```

```

using (Document doc = new Document())
{
    // Add registration keys
    doc.RegistrationName = "demo";
    doc.RegistrationKey = "demo";

    // Load document
    doc.Load("sample.pdf");

    // Write template in each pages
    for (int i = 0; i < doc.Pages.Count; i++)
    {
        Page currentPage = doc.Pages[i];
        _AddPageNo(currentPage, "Page #" + (i + 1),
PageNoStyle.BottomRight);
    }

    // Save output file
    doc.Save("result.pdf");
}

// Open result document in default associated application (for demo
purpose)
ProcessStartInfo processStartInfo = new
ProcessStartInfo("result.pdf");
processStartInfo.UseShellExecute = true;
Process.Start(processStartInfo);
}
catch (Exception ex)
{
    Console.WriteLine(ex.Message);
    Console.WriteLine("Press enter key to exit...");
    Console.ReadLine();
}
}

///
/// Add Page No
///

private static void _AddPageNo(Page currentPage, string pageNoText,
PageNoStyle pageNoStyle)
{
    float float_left = 10;
    float float_top = 10;

    switch (pageNoStyle)
    {
        case pageNoStyle.TopRight:
            float_left = currentPage.Width - 60;
            break;
        case pageNoStyle.BottomLeft:
            float_top = currentPage.Height - 20;
            break;
        case pageNoStyle.BottomRight:
            float_top = currentPage.Height - 20;
            float_left = currentPage.Width - 60;
            break;
        case pageNoStyle.TopLeft:

```

```
        default:
            break;
    }

    // Draw/Write page no
    currentPage.Canvas.DrawString(pageNoText,
        new Font(StandardFonts.CourierBold, 12),
        new SolidBrush(),
        float_left,
        float_top);
    }
}
}
```

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](http://www.ByteScout.com)

or

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