www.bytescout.com

hide images in PDF document in C# using ByteScout Premium Suite

hide images in PDF document in C#

The example source codes on this page will display you how to make hide images in PDF document in C#. ByteScout Premium Suite was created to assist hide images in PDF document in C#. ByteScout Premium Suite is the bundle that includes twelve SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording.

Want to speed up the application development? Then this C#, code samples for C#, developers help to speed up the application development and writing a code when using ByteScout Premium Suite. Just copy and paste this C# sample code to your C# application's code editor, add a reference to ByteScout Premium Suite (if you haven't added yet) and you are ready to go! Want to see how it works with your data then code testing will allow the function to be tested and work properly.

Trial version along with the source code samples for C# can be downloaded from our website

FOR MORE INFORMATION AND FREE TRIAL:

Download Free Trial SDK (on-premise version)

Read more about ByteScout Premium Suite

Explore API Documentation

Get Free Training for ByteScout Premium Suite

Get Free API key for Web API

visit www.ByteScout.com

Source Code Files:

HideImagesInPdfDocument.sln

```
Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio Version 16
VisualStudioVersion = 16.0.28729.10
MinimumVisualStudioVersion = 10.0.40219.1
Project("{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}") = "HideImagesInPdfDocument", "HideIm
EndProject
Global
        GlobalSection(SolutionConfigurationPlatforms) = preSolution
                Debug|Any CPU = Debug|Any CPU
                Release|Any CPU = Release|Any CPU
        EndGlobalSection
        GlobalSection(ProjectConfigurationPlatforms) = postSolution
                {AF6915BA-85D2-43E3-AD28-2743B7E805B1}.Debug|Any CPU.ActiveCfg = Debug
                {AF6915BA-85D2-43E3-AD28-2743B7E805B1}.Debug|Any CPU.Build.0 = Debug|A
                {AF6915BA-85D2-43E3-AD28-2743B7E805B1}.Release|Any CPU.ActiveCfg = Rel
                {AF6915BA-85D2-43E3-AD28-2743B7E805B1}.Release|Any CPU.Build.0 = Releas
        EndGlobalSection
        GlobalSection(SolutionProperties) = preSolution
                HideSolutionNode = FALSE
        EndGlobalSection
        GlobalSection(ExtensibilityGlobals) = postSolution
                SolutionGuid = {3BF0F1FD-BA8E-45F6-ABB2-F7EB20F38CEF}
        EndGlobalSection
EndGlobal
```

Program.cs

```
using System.Diagnostics;
using System.Drawing;
using Bytescout.PDF;
using Bytescout.PDFExtractor;
using Image = Bytescout.PDF.Image;


namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            string inputPdfDocument = @".\sample.pdf";
            string maskingImage = @".\mask.png";
            string resultPdfDocument = @".\result.pdf";


            // Create Bytescout.PDFExtractor.ImageExtractor instance
            using (ImageExtractor imageExtractor = new ImageExtractor())
```

```
            {
                imageExtractor.RegistrationName = "demo";
                imageExtractor.RegistrationKey = "demo";

                // Load document
                imageExtractor.LoadDocumentFromFile(inputPdfDocument);

                // Create Bytescout.PDF.Document instance
                using (Document pdfDocument = new Document())
                {
                    pdfDocument.RegistrationName = "demo";
                    pdfDocument.RegistrationKey = "demo";

                    // Load document
                    pdfDocument.Load(inputPdfDocument);

                    // Load masking image
                    var maskImage = new Image(maskingImage);

                    // Process pages
                    for (int i = 0; i < pdfDocument.Pages.Count; i++)
                    {
                        Page page = pdfDocument.Pages[i];

                        // Find images on the page
                        if (imageExtractor.GetFirstPageImage(i))
                        {
                            do
                            {
                                // Get image rectangle
                                var imageRect = imageExtractor.GetCurrentImageRectangle

                                // Draw new image other the found image
                                page.Canvas.DrawImage(maskImage, imageRect.Left, imageR

                            } while (imageExtractor.GetNextImage());
                        }
                    }

                    // Save modified document
                    pdfDocument.Save(resultPdfDocument);
                }
            }

            // Open document in default PDF viewer application (for demonstration).
            Process.Start(resultPdfDocument);
        }
    }
}
```

VIDEO

https://www.youtube.com/watch?v=NEwNs2b9YN8

ON-PREMISE OFFLINE SDK

60 Day Free Trial or Visit ByteScout Premium Suite Home Page
Explore ByteScout Premium Suite Documentation
Explore Samples
Sign Up for ByteScout Premium Suite 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