

How to convert scanned PDF to excel in C# with ByteScout PDF Suite

If you want to learn more then this tutorial will show how to convert scanned PDF to excel in C#

Source code documentation samples give simple and easy method to install a needed feature into your application. ByteScout PDF Suite is the bundle that provides six different SDK libraries to work with PDF from generating rich PDF reports to extracting data from PDF documents and converting them to HTML. This bundle includes PDF (Generator) SDK, PDF Renderer SDK, PDF Extractor SDK, PDF to HTML SDK, PDF Viewer SDK and PDF Generator SDK for Javascript. It can convert scanned PDF to excel in C#.

Want to save time? You will save a lot of time on writing and testing code as you may just take the C# code from ByteScout PDF Suite for convert scanned PDF to excel below and use it in your application. Simply copy and paste in your C# project or application you and then run your app! Further improvement of the code will make it more robust.

ByteScout provides the free trial version of ByteScout PDF Suite along with the documentation and source code samples.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout PDF Suite](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout PDF Suite](#)

[Get Free API key for Web API](#)

[visit www.Bytescout.com](http://www.Bytescout.com)

Source Code Files:

```

namespace Sample_UI_Application
{
    partial class Form1
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            System.ComponentModel.ComponentResourceManager resources = new System.ComponentModel.ComponentResourceManager(typeof(Form1));
            this.pdfViewerControl1 = new Bytescout.PDFViewer.PDFViewerControl();
            this.toolStrip1 = new System.Windows.Forms.ToolStrip();
            this.tsbOpen = new System.Windows.Forms.ToolStripButton();
            this.toolStripSeparator1 = new System.Windows.Forms.ToolStripSeparator();
            this.tsbExportToCSV = new System.Windows.Forms.ToolStripButton();
            this.tsbExportToXLSX = new System.Windows.Forms.ToolStripButton();
            this.toolStrip1.SuspendLayout();
            this.SuspendLayout();
            //
            // pdfViewerControl1
            //
            this.pdfViewerControl1.BackColor = System.Drawing.SystemColors.ButtonShadow;
            this.pdfViewerControl1.Dock = System.Windows.Forms.DockStyle.Fill;
            this.pdfViewerControl1.Location = new System.Drawing.Point(0, 25);
            this.pdfViewerControl1.MouseMode = Bytescout.PDFViewer.MouseMode.Selection;
            this.pdfViewerControl1.Name = "pdfViewerControl1";
            this.pdfViewerControl1.RegistrationKey = null;
            this.pdfViewerControl1.RegistrationName = null;
            this.pdfViewerControl1.ResetRotationOnPageChange = false;
            this.pdfViewerControl1.Scale = 100;
            this.pdfViewerControl1.SelectionColor = System.Drawing.Color.Red;
            this.pdfViewerControl1.ShowImageObjects = true;
            this.pdfViewerControl1.ShowTextObjects = true;
            this.pdfViewerControl1.ShowVectorObjects = true;
        }
    }
}

```

```

this.pdfViewerControl1.Size = new System.Drawing.Size(842, 514);
this.pdfViewerControl1.TabIndex = 0;
//
// toolStrip1
//
this.toolStrip1.Items.AddRange(new System.Windows.Forms.ToolStripItem[] {
this.tsbOpen,
this.toolStripSeparator1,
this.tsbExportToCSV,
this.tsbExportToXLSX});
this.toolStrip1.Location = new System.Drawing.Point(0, 0);
this.toolStrip1.Name = "toolStrip1";
this.toolStrip1.Size = new System.Drawing.Size(842, 25);
this.toolStrip1.TabIndex = 1;
this.toolStrip1.Text = "toolStrip1";
//
// tsbOpen
//
this.tsbOpen.Image = global::Sample_UI_Application.Properties.Resources.fot;
this.tsbOpen.ImageTransparentColor = System.Drawing.Color.Magenta;
this.tsbOpen.Name = "tsbOpen";
this.tsbOpen.Size = new System.Drawing.Size(80, 22);
this.tsbOpen.Text = "&Open PDF";
this.tsbOpen.Click += new System.EventHandler(this.tsbOpen_Click);
//
// toolStripSeparator1
//
this.toolStripSeparator1.Name = "toolStripSeparator1";
this.toolStripSeparator1.Size = new System.Drawing.Size(6, 25);
//
// tsbExportToCSV
//
this.tsbExportToCSV.Image = ((System.Drawing.Image)(resources.GetObject("tsbExportToCSV.Image")));
this.tsbExportToCSV.ImageTransparentColor = System.Drawing.Color.Magenta;
this.tsbExportToCSV.Name = "tsbExportToCSV";
this.tsbExportToCSV.Size = new System.Drawing.Size(100, 22);
this.tsbExportToCSV.Text = "Export To CSV";
this.tsbExportToCSV.Click += new System.EventHandler(this.tsbExportToCSV_Click);
//
// tsbExportToXLSX
//
this.tsbExportToXLSX.Image = ((System.Drawing.Image)(resources.GetObject("tsbExportToXLSX.Image")));
this.tsbExportToXLSX.ImageTransparentColor = System.Drawing.Color.Magenta;
this.tsbExportToXLSX.Name = "tsbExportToXLSX";
this.tsbExportToXLSX.Size = new System.Drawing.Size(105, 22);
this.tsbExportToXLSX.Text = "Export To XLSX";
this.tsbExportToXLSX.Click += new System.EventHandler(this.tsbExportToXLSX_Click);
//
// Form1
//
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(842, 539);
this.Controls.Add(this.pdfViewerControl1);
this.Controls.Add(this.toolStrip1);
this.Name = "Form1";
this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;
this.Text = "Form1";
this.toolStrip1.ResumeLayout(false);
this.toolStrip1.PerformLayout();

```

```

        this.ResumeLayout(false);
        this.PerformLayout();

    }

    #endregion

    private Bytescout.PDFViewer.PDFViewerControl pdfViewerControl1;
    private System.Windows.Forms.ToolStrip toolStrip1;
    private System.Windows.Forms.ToolStripButton tsbOpen;
private System.Windows.Forms.ToolStripSeparator toolStripSeparator1;
private System.Windows.Forms.ToolStripButton tsbExportToCSV;
private System.Windows.Forms.ToolStripButton tsbExportToXLSX;
    }
}

```

Form1.cs

```

using System;
using System.Diagnostics;
using System.Drawing;
using System.Windows.Forms;
using Bytescout.PDFExtractor;

namespace Sample_UI_Application
{
    // This example requires 'PDF Viewer SDK' and 'PDF Extractor SDK' installed.
    // Download link: http://cdn.bytescout.com/ByteScoutInstaller.exe

    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        protected override void OnLoad(EventArgs e)
        {
            // Preload document into viewer
            pdfViewerControl1.InputFile = @".\sample_ocr.pdf";

            base.OnLoad(e);
        }

        private void tsbOpen_Click(object sender, EventArgs e)
        {
            using (OpenFileDialog openFileDialog = new OpenFileDialog())
            {
                openFileDialog.Title = @"Open PDF Document";
            }
        }
    }
}

```

```

        openFileDialog.Filter = @"PDF Files (*.pdf)|*.pdf|All Files (*.*)|*.*";

        if (openFileDialog.ShowDialog() == DialogResult.OK)
        {
            this.Text = openFileDialog.FileName;

            Cursor = Cursors.WaitCursor;

            try
            {
                pdfViewerControl1.InputFile = openFileDialog.FileName;
            }
            catch (Exception exception)
            {
                MessageBox.Show(exception.Message);
            }
            finally
            {
                Cursor = Cursors.Default;
            }
        }
    }

    private void tsbExportToCSV_Click(object sender, EventArgs e)
    {
        // Get selections from viewer
        RectangleF[] selections = pdfViewerControl1.SelectionInPoints;

        string outputFile = @".\result.csv";

        using (CSVExtractor csvExtractor = new CSVExtractor("demo", "demo"))
        {
            // Load document into extractor
            csvExtractor.LoadDocumentFromFile(pdfViewerControl1.InputFile);

            // Enable OCR to recognize text from images
            csvExtractor.OCRMode = OCRMode.Auto;
            csvExtractor.OCRResolution = 300;
            csvExtractor.OCRLanguage = "eng";
            csvExtractor.OCRLanguageDataFolder = @"c:\Program Files\Bytescout PDF Extractor";

            // There are double spaces between some words in your document.
            // To avoid such words break column structure increase the space ratio
            csvExtractor.DetectNewColumnBySpacesRatio = 2;

            // FYI, removing horizontal lines may increase the text recognition quality
            csvExtractor.OCRImagePreprocessingFilters.AddHorizontalLinesRemover();
            // Another filter able to improve the recognition
            //csvExtractor.OCRImagePreprocessingFilters.AddGammaCorrection();

            // If selection exists set the extraction area.
            // Otherwise it will extract the whole page.
            if (selections.Length > 0)
                csvExtractor.SetExtractionArea(selections[0]);

            // Save extraction results to CSV files
            csvExtractor.SavePageCSVToFile(pdfViewerControl1.CurrentPageIndex, outputFile);
        }
    }
}

```

```

        Process.Start(outputFile);
    }

    private void tsbExportToXLSX_Click(object sender, EventArgs e)
    {
        // Get selections from viewer
        RectangleF[] selections = pdfViewerControl1.SelectionInPoints;

        string outputFile = @".\result.xlsx";

        using (XLSExtractor xlsExtractor = new XLSExtractor("demo", "demo"))
        {
            // Load document into extractor
            xlsExtractor.LoadDocumentFromFile(pdfViewerControl1.InputFile);

            xlsExtractor.OCRMode = OCRMode.Auto;
            xlsExtractor.OCRResolution = 300;
            xlsExtractor.OCRLanguage = "eng";
            xlsExtractor.OCRLanguageDataFolder = @"c:\Program Files\Bytescout PDF I

            xlsExtractor.OutputFormat = SpreadseetOutputFormat.XLSX;
            xlsExtractor.RichTextFormatting = false;

            // There are double spaces between some words in your document.
            // To avoid such words break column structure increase the space ratio
            xlsExtractor.DetectNewColumnBySpacesRatio = 2;

            // FYI, removing horizontal lines may increase the text recognition qu
            //xlsExtractor.OCRImagePreprocessingFilters.AddHorizontalLinesRemover()
            // Another filter able to improve the recognition
            //xlsExtractor.OCRImagePreprocessingFilters.AddGammaCorrection();

            // If selection exists set the extraction area.
            // Otherwise it will extract the whole page.
            if (selections.Length > 0)
                xlsExtractor.SetExtractionArea(selections[0]);

            // Save extraction results to XLSX files
            xlsExtractor.SavePageToXLSFile(pdfViewerControl1.CurrentPageIndex, outp
        }

        Process.Start(outputFile);
    }
}
}
}

```

Program.cs

```

using System;
using System.Collections.Generic;

```

```

using System.Windows.Forms;

namespace Sample_UI_Application
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}

```

ScannedPdfToExcel.sln

```

Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio 15
VisualStudioVersion = 15.0.27130.2027
MinimumVisualStudioVersion = 10.0.40219.1
Project("{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}") = "ScannedPdfToExcel", "ScannedPdfToExcel.sln"
EndProject
Global
    GlobalSection(SolutionConfigurationPlatforms) = preSolution
        Debug|Any CPU = Debug|Any CPU
        Release|Any CPU = Release|Any CPU
    EndGlobalSection
    GlobalSection(ProjectConfigurationPlatforms) = postSolution
        {0DB74CC3-1DD8-4A58-94FC-CA9A60E2F8A2}.Debug|Any CPU.ActiveCfg = Debug|Any CPU
        {0DB74CC3-1DD8-4A58-94FC-CA9A60E2F8A2}.Debug|Any CPU.Build.0 = Debug|Any CPU
        {0DB74CC3-1DD8-4A58-94FC-CA9A60E2F8A2}.Release|Any CPU.ActiveCfg = Release|Any CPU
        {0DB74CC3-1DD8-4A58-94FC-CA9A60E2F8A2}.Release|Any CPU.Build.0 = Release|Any CPU
    EndGlobalSection
    GlobalSection(SolutionProperties) = preSolution
        HideSolutionNode = FALSE
    EndGlobalSection
    GlobalSection(ExtensibilityGlobals) = postSolution
        SolutionGuid = {4701A661-4DCD-4CAC-9E81-162B4D0AB70B}
    EndGlobalSection
EndGlobal

```

app.config

```
<?xml version="1.0" encoding="utf-8"?>  
<configuration>  
<startup><supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.0"/></startup>
```

VIDEO

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

ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout PDF Suite Home Page](#)
[Explore ByteScout PDF Suite Documentation](#)
[Explore Samples](#)
[Sign Up for ByteScout PDF Suite Online Training](#)

ON-DEMAND REST WEB API

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

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

[visit www.PDF.co](http://www.PDF.co)

www.bytescout.com