

## How to convert rtf email to pdf with pdf sdk in C# and ByteScout Premium Suite

If you want to learn more then this tutorial will show how to convert rtf email to pdf with pdf sdk in C#

Every ByteScout tool includes simple example C# source codes that you can get here or in the folder with installed ByteScout product. ByteScout Premium Suite can convert rtf email to pdf with pdf sdk. It can be applied from 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 quickly learn? This fast application programming interfaces of ByteScout Premium Suite for C# plus the guidelines and the code below will help you quickly learn how to convert rtf email to pdf with pdf sdk. Simply copy and paste in your C# project or application you and then run your app! Want to see how it works with your data then code testing will allow the function to be tested and work properly.

You can download free trial version of ByteScout Premium Suite 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 Premium Suite](#)

[Explore API Documentation](#)

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

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

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

Source Code Files:

```
Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio 15
VisualStudioVersion = 15.0.27703.2026
MinimumVisualStudioVersion = 10.0.40219.1
Project("{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}") = "EmailToPDF_RTFEmail", "EmailToPDF
EndProject
Global
    GlobalSection(SolutionConfigurationPlatforms) = preSolution
        Debug|Any CPU = Debug|Any CPU
        Release|Any CPU = Release|Any CPU
    EndGlobalSection
    GlobalSection(ProjectConfigurationPlatforms) = postSolution
        {8B33CB1C-B6A2-4750-9D1C-EB963DC8A17D}.Debug|Any CPU.ActiveCfg = Debug
        {8B33CB1C-B6A2-4750-9D1C-EB963DC8A17D}.Debug|Any CPU.Build.0 = Debug|Any CPU
        {8B33CB1C-B6A2-4750-9D1C-EB963DC8A17D}.Release|Any CPU.ActiveCfg = Release
        {8B33CB1C-B6A2-4750-9D1C-EB963DC8A17D}.Release|Any CPU.Build.0 = Release
    EndGlobalSection
    GlobalSection(SolutionProperties) = preSolution
        HideSolutionNode = FALSE
    EndGlobalSection
    GlobalSection(ExtensibilityGlobals) = postSolution
        SolutionGuid = {88E4CD5D-0D2E-42FD-BD52-BFF59A042E25}
    EndGlobalSection
EndGlobal
```

## Program.cs

```
using System;
using System.Diagnostics;
using System.Drawing.Printing;
using System.IO;
using System.Text;
using Bytescout.PDF;
using Bytescout.PDF.Converters;

using Font = Bytescout.PDF.Font;
using SolidBrush = Bytescout.PDF.SolidBrush;

namespace EmailToPDF_HTMLEmail
{
    class Program
    {
        static void Main(string[] args)
        {
            try
```

```

{
    // Parse MessageContents using MsgReader Library
    // MsgReader library can be obtained from: https://github.com/Sicos1977/MsgReader
    using (var msg = new MsgReader.Outlook.Storage.Message("RtfSampleEmail.rtf"))
    {
        // Get Sender information
        var from = msg.GetEmailSender(false, false);

        // Message sent datetime
        var sentOn = msg.SentOn;

        // Recipient To information
        var recipientsTo = msg.GetEmailRecipients(MsgReader.Outlook.RecipientType.To);

        // Recipient CC information
        var recipientsCc = msg.GetEmailRecipients(MsgReader.Outlook.RecipientType.Cc);

        // Message subject
        var subject = msg.Subject;

        // Get Message Body
        var msgBody = msg.BodyRtf;

        // Convert RTF to HTML
        msgBody = RtfPipe.Rtf.ToHtml(msgBody);

        // Prepare PDF document
        using (Document outputDocument = new Document())
        {
            // Add registration keys
            outputDocument.RegistrationName = "demo";
            outputDocument.RegistrationKey = "demo";

            // Add page
            Page page = new Page(PaperFormat.A4);
            outputDocument.Pages.Add(page);

            // Add sample content
            Font font = new Font(StandardFonts.Times, 12);
            Brush brush = new SolidBrush();

            // Add Email contents
            int topMargin = 20;
            page.Canvas.DrawString("File Name: {msg.FileName}", font, brush, 20, topMargin);
            page.Canvas.DrawString("From: {from}", font, brush, 20, topMargin + 20);
            page.Canvas.DrawString("Sent On: {(sentOn.HasValue ? sentOn.ToString() : null)}", font, brush, 20, topMargin + 40);
            page.Canvas.DrawString("To: {recipientsTo}", font, brush, 20, topMargin + 60);

            if (!string.IsNullOrEmpty(recipientsCc))
            {
                page.Canvas.DrawString("CC: {recipientsCc}", font, brush, 20, topMargin + 80);
            }

            page.Canvas.DrawString("Subject: {subject}", font, brush, 20, topMargin + 100);
            page.Canvas.DrawString("Message body in next page.", font, brush, 20, topMargin + 120);

            // Convert Html body to PDF in order to retain all formatting.
            using (HtmlToPdfConverter converter = new HtmlToPdfConverter())

```

```

    {
        converter.PageSize = PaperKind.A4;
        converter.Orientation = Bytescout.PDF.Converters.PaperOrient

        // Convert input HTML to stream
        byte[] byteArrayBody = Encoding.UTF8.GetBytes(msgBody);
        MemoryStream inputStream = new MemoryStream(byteArrayBody)

        // Create output stream to store generated PDF file
        using (var outputStream = new MemoryStream())
        {
            // Convert HTML to PDF
            converter.ConvertHtmlToPdf(inputStream, outputStream);

            // Create new document from generated output stream
            Document docContent = new Document(outputStream);

            // Append all pages to main PDF
            foreach (Page item in docContent.Pages)
            {
                outputDocument.Pages.Add(item);
            }

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

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

```

packages.config

```

<?xml version="1.0" encoding="utf-8"?>
<packages>
  <package id="MsgReader" version="3.4.0" targetFramework="net45" />

```

```
<package id="OpenMcdf" version="2.2.1.3" targetFramework="net45" />  
<package id="RtfPipe" version="1.0.0.23" targetFramework="net45" />  
</packages>
```

---

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

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

[www.bytescout.com](http://www.bytescout.com)