

How to reduce memory usage during PDF to HTML conversion in C# and ByteScout PDF To HTML SDK

This tutorial will show how to reduce memory usage during PDF to HTML conversion in C#

The coding tutorials are designed to help you test the features without need to write your own code. ByteScout PDF To HTML SDK is the SDK that can take PDF and generate HTML version of it with all the visual layout, positions, vectors, images and form fields preserved. Generated HTML requires no special software and can be viewed in any Internet browser. It can reduce memory usage during PDF to HTML conversion in C#.

C# code samples for C# developers help to speed up coding of your application when using ByteScout PDF To HTML SDK. In your C# project or application you may simply copy & paste the code and then run your app! Further enhancement of the code will make it more vigorous.

Download free trial version of ByteScout PDF To HTML SDK from our website with this and other source code samples for C#.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout PDF To HTML SDK](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout PDF To HTML SDK](#)

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

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

Source Code Files:

```

using System;
using System.Diagnostics;
using Bytestout.PDF2HTML;

namespace ReduceMemoryUsage
{
    class Program
    {
        static void Main(string[] args)
        {
            // When processing huge PDF documents you may run into OutOfMemoryException
            // This example demonstrates a way to spare the memory by disabling page data caching

            // Create Bytestout.PDF2HTML.HTMLExtractor instance
            using (HTMLExtractor extractor = new HTMLExtractor("demo", "demo"))
            {
                try
                {
                    // Load sample PDF document
                    extractor.LoadDocumentFromFile("sample2.pdf");

                    // Disable page data caching, so processed pages will be disposed of
                    extractor.PageDataCaching = PageDataCaching.None;

                    // Save result to file
                    extractor.SaveHtmlToFile("output.html");
                }
                catch (PDF2HTMLException exception)
                {
                    Console.WriteLine(exception.ToString());
                }
            }

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

```

VIDEO

https://www.youtube.com/watch?v=kB2ELm_kCds

ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout PDF To HTML SDK Home Page](#)
[Explore ByteScout PDF To HTML SDK Documentation](#)
[Explore Samples](#)
[Sign Up for ByteScout PDF To HTML SDK 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