

auto redact text from PDF by keywords in C# and ByteScout Sensitive Data Suite

Learn to code auto redact text from PDF by keywords in C#: How-To tutorial

An easy to understand guide to learn how to auto redact text from PDF by keywords in C#. ByteScout Sensitive Data Suite helps with auto redact text from PDF by keywords in C#. ByteScout Sensitive Data Suite is the bundle that includes multiple components from ByteScout for working with sensitive and personal data. With these components you may analyze, redact, remove, blackout sensitive data in documents and pdf.

The below SDK samples describe how to quickly make your application do auto redact text from PDF by keywords in C# with the help of ByteScout Sensitive Data Suite. To use auto redact text from PDF by keywords in your C# project or application just copy & paste the code and then run your app! Enjoy writing a code with ready-to-use sample C# codes to implement auto redact text from PDF by keywords using ByteScout Sensitive Data Suite.

ByteScout Sensitive Data Suite free trial version is available for download from our website. Free trial also includes programming tutorials along with source code samples.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout Sensitive Data Suite](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout Sensitive Data Suite](#)

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

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

Source Code Files:

```
using System;
using System.Drawing;
using Bytestcout.PDFExtractor;

namespace AutoRedactTextFromPDFByKeywords
{
    class Program
    {
        static void Main(string[] args)
        {
            // Create Bytestcout.PDFExtractor.Remover instance
            Remover remover = new Remover("demo", "demo");

            // Load sample PDF document
            remover.LoadDocumentFromFile(@"sample1.pdf");

            // Search Keyword
            string SearchKeyword = "Martian dichotomy";

            // Prepare TextExtractor
            using (TextExtractor textExtractor = new TextExtractor("demo", "demo"))
            {
                // Load document into TextExtractor
                textExtractor.LoadDocumentFromFile(@"sample1.pdf");

                // Set word matching options
                textExtractor.WordMatchingMode = WordMatchingMode.None;

                ISearchResult[] searchResults = textExtractor.FindAll(0, SearchKeyword);

                // Remove text objects find by SearchResults.
                // NOTE: The removed text might be larger than the specified rectangle
                // to split PDF text objects.
                remover.RemoveText(searchResults, @"result1.pdf");
            }

            // Open output file in default application
            System.Diagnostics.Process.Start("result1.pdf");

            // Clean up.
            remover.Dispose();
        }
    }
}
```

VIDEO

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

ON-PREMISE OFFLINE SDK

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