

How to reduce memory usage with pdf extractor sdk in VB.NET using ByteScout Data Extraction Suite

Learn to code in VB.NET to reduce memory usage with pdf extractor sdk with this step-by-step tutorial

The code displayed below will guide you to install an VB.NET app to reduce memory usage with pdf extractor sdk. ByteScout Data Extraction Suite can reduce memory usage with pdf extractor sdk. It can be applied from VB.NET. ByteScout Data Extraction Suite is the bundle that includes three SDK tools for data extraction from PDF, scans, images and from spreadsheets: PDF Extractor SDK, Data Extraction SDK, Barcode Reader SDK.

This prolific sample source code in VB.NET for ByteScout Data Extraction Suite contains various functions and other necessary options you should do calling the API to reduce memory usage with pdf extractor sdk. This VB.NET sample code is all you need for your app. Just copy and paste the code, add references (if needs to) and you are all set! This basic programming language sample code for VB.NET will do the whole work for you to reduce memory usage with pdf extractor sdk.

ByteScout provides the free trial version of ByteScout Data Extraction 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 Data Extraction Suite](#)

[Explore API Documentation](#)

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

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

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

Source Code Files:

Program.vb

```
Imports Bytescout.PDFExtractor
Imports System.IO

Class Program

    Friend Shared Sub Main(args As String())

        ' 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 Bytescout.PDFExtractor.TextExtractor instance
        Using extractor As New TextExtractor("demo", "demo")

            Try

                ' Load sample PDF document
                extractor.LoadDocumentFromFile("sample2.pdf")

                ' Disable page data caching, so processed pages will be disposed automatically
                extractor.PageDataCaching = PageDataCaching.None

                ' Save extracted text to file
                extractor.SaveTextToFile("output.txt")

                Catch exception As PDFExtractorException
                    Console.WriteLine(exception.ToString())
                End Try

            End Using

            ' Open result file in default associated application (for demo purposes)
            System.Diagnostics.Process.Start("output.txt")

        End Sub

    End Class
```

VIDEO

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

ON-PREMISE OFFLINE SDK

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