

How to find keyword in pdf and extract page with pdf extractor sdk in VB.NET and ByteScout Premium Suite

This code in VB.NET shows how to find keyword in pdf and extract page with pdf extractor sdk with this how to tutorial

These source code samples are assembled by their programming language and functions they apply. ByteScout Premium Suite is the set that includes 12 SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording. It can find keyword in pdf and extract page with pdf extractor sdk in VB.NET.

Want to save time? You will save a lot of time on writing and testing code as you may just take the VB.NET code from ByteScout Premium Suite for find keyword in pdf and extract page with pdf extractor sdk below and use it in your application. IF you want to implement the functionality, just copy and paste this code for VB.NET below into your code editor with your app, compile and run your application. Further improvement of the code will make it more robust.

The trial version of ByteScout Premium Suite can be downloaded for free from our website. It also includes source code samples for VB.NET and other programming languages.

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:

```

' This example page extraction by found keyword.

Imports Bytescout.PDFExtractor

Class Program

    Friend Shared Sub Main(args As String())

        Dim inputFile As String = ".\sample2.pdf"

        ' Create Bytescout.PDFExtractor.TextExtractor instance
        Dim extractor As New TextExtractor()
        extractor.RegistrationName = "demo"
        extractor.RegistrationKey = "demo"

        ' Load sample PDF document
        extractor.LoadDocumentFromFile(inputFile)

        Dim pageCount As Integer = extractor.GetPageCount()

        ' Search each page for a keyword
        For i As Integer = 0 To pageCount - 1

            If extractor.Find(i, "bombardment", False) Then

                ' Extract page
                Using splitter As New DocumentSplitter("demo", "demo")

                    splitter.OptimizeSplittedDocuments = True

                    Dim pageNumber As Integer = i + 1
                    ' (!) page number in ExtractPage() is 1-based
                    Dim outputFile As String = ".\page" & pageNumber.ToString() & ".pdf"
                    splitter.ExtractPage(inputFile, outputFile, pageNumber)

                    Console.WriteLine("Extracted page " & pageNumber.ToString() & " to " & outputFile)

                End Using
            End If
        Next

        ' Cleanup
        extractor.Dispose()

        Console.WriteLine()
        Console.WriteLine("Press any key...")
        Console.ReadKey()

    End Sub
End Class

```

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