

How to split pdf document by found barcode with barcode reader sdk in VB.NET with ByteScout Premium Suite

If you want to learn more then this tutorial will show how to split pdf document by found barcode with barcode reader sdk in VB.NET

We made thousands of pre-made source code pieces for easy implementation in your own programming projects. ByteScout Premium Suite is the set that includes 12 SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording and you can use it to split pdf document by found barcode with barcode reader sdk with VB.NET.

The following code snippet for ByteScout Premium Suite works best when you need to quickly split pdf document by found barcode with barcode reader sdk in your VB.NET application. Follow the instructions from scratch to work and copy the VB.NET code. Complete and detailed tutorials and documentation are available along with installed ByteScout Premium Suite if you'd like to learn more about the topic and the details of the API.

If you want to try other source code samples then the free trial version of ByteScout Premium Suite is available for download from our website. Just try other source code samples for VB.NET.

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:

```

Imports System.IO
Imports System.Text
Imports Bytescout.BarCodeReader

Module Module1

    Sub Main()

        Const inputFile As String = "Barcodes.pdf"

        Console.WriteLine("Reading barcode(s) from image {0}", Path.GetFullPath(inputFile))

        ' Create Bytescout.BarCodeReader.Reader instance
        Dim reader As New Reader()
        reader.RegistrationName = "demo"
        reader.RegistrationKey = "demo"

        ' Set barcode type to find
        reader.BarcodeTypesToFind.Code39 = True

        ' -----
        ' NOTE: We can read barcodes from specific page to increase performance .
        ' For sample please refer to "Decoding barcodes from PDF by pages" program.
        ' -----

        ' Find barcode in PDF document
        reader.ReadFrom(inputFile)

        ' Method 1: Split PDF document in two parts by found barcode
        ' NOTE: In Full version of the SDK this method is unlocked in "PRO" license type
        reader.SplitDocument("barcodes.pdf", "part1.pdf", "part2.pdf", reader.FoundBarcodes)

        ' Method 2: Extract page containing the barcode from PDF document
        ' NOTE: In Full version of the SDK this method is unlocked in "PRO" license type
        reader.ExtractPageFromDocument("barcodes.pdf", "extracted_page.pdf", reader.FoundBarcodes)

        ' Method 3: Split PDF document into parts in one pass.
        ' NOTE: In Full version of the SDK this method is unlocked in "PRO" license type

        Dim pageRanges As StringBuilder = New StringBuilder()

        ' Create string containing page ranges to extract in the form "1-4,6-8,10-11,12"
        For i As Integer = 0 To reader.FoundBarcodes.Length - 1

            Dim barcode As FoundBarcode = reader.FoundBarcodes(i)

            ' Add pages before the first barcode found
            If i = 0 And barcode.Page > 0 Then
                pageRanges.Append("1")
                If barcode.Page > 1 Then

```

```

        pageRanges.Append("-")
        pageRanges.Append(barcode.Page)
    End If
    pageRanges.Append(",")
End If

' Add page with barcode
pageRanges.Append(barcode.Page + 1) ' +1 because we skip the page with barcode

' Add range untill the next barcode
If i < reader.FoundBarcodes.Length - 1 Then
    If reader.FoundBarcodes(i + 1).Page - barcode.Page > 1 Then
        pageRanges.Append("-")
        pageRanges.Append(reader.FoundBarcodes(i + 1).Page)
    End If
    pageRanges.Append(",")
Else
    ' for the last found barcode add ending "-" meaning "to the last page"
    pageRanges.Append("-")
End If
Next

' Split document
Dim splittedParts As String() = reader.SplitDocument("barcodes.pdf", pageRanges)
' The method returns array of file names. Rename files as desired.

' Cleanup
reader.Dispose()

For Each fileName As String In splittedParts
    Console.WriteLine(fileName)
Next

Console.WriteLine("Press enter key to exit...")
Console.ReadLine()

End Sub

End Module

```

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