

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

Step-by-step tutorial on how to split pdf document by found barcode with barcode reader sdk in VB.NET

These source code samples are assembled by their programming language and functions they apply. 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. It can split pdf document by found barcode with barcode reader sdk in VB.NET.

The SDK samples given below describe how to quickly make your application do split pdf document by found barcode with barcode reader sdk in VB.NET with the help of ByteScout Data Extraction Suite. Simply copy and paste in your VB.NET project or application you and then run your app! Enjoy writing a code with ready-to-use sample VB.NET codes.

The trial version of ByteScout Data Extraction 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 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:

```

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 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 \[www.PDF.co\]\(http://www.PDF.co\)](#)

[\[www.bytescout.com\]\(http://www.bytescout.com\)](#)