

How to extract pages from PDF in VB.NET using ByteScout PDF Extractor SDK

Write code in VB.NET to extract pages from PDF with this step-by-step tutorial

The sample shows steps and algorithm of how to extract pages from PDF and how to make it work in your VB.NET application. ByteScout PDF Extractor SDK is the SDK is designed to help developers with pdf tables and pdf data extraction from unstructured documents like pdf, tiff, scans, images, scanned and electronic forms. The library is powered by OCR, computer vision and AI to provide unique functionality like table detection, automatic table structure extraction, data restoration, data restructuring and reconstruction. Supports PDF, TIFF, PNG, JPG images as input and can output CSV, XML, JSON formatted data. Includes full set of utilities like pdf splitter, pdf merger, searchable pdf maker and other utilities. It can be used to extract pages from PDF using VB.NET.

This rich sample source code in VB.NET for ByteScout PDF Extractor SDK includes the number of functions and options you should do calling the API to extract pages from PDF. Follow the instructions from the scratch to work and copy the VB.NET code. Further enhancement of the code will make it more vigorous.

Download free trial version of ByteScout PDF Extractor SDK from our website with this and other source code samples for VB.NET.

VB.NET - Program.vb

```
' This example demonstrates various PDF document splitting scenarios:
' - extract a single page;
' - split in two parts;
' - split by ranges specified in text form: "1-5,6,7-10,11-".

Imports System.IO
Imports Bytescout.PDFExtractor

Class Program
    Friend Shared Sub Main(args As String())
        Dim inputFile As String = ".\sample.pdf"

        Using splitter As New DocumentSplitter("demo", "demo")
            splitter.OptimizeSplittedDocuments = True

            ' Extracting specific page:
            ' =====

            splitter.ExtractPage(inputFile, "page3.pdf", 3)
            ' (!) Note: page number is 1-based.
            Console.WriteLine("Extracted page 3 to file ""page3.pdf""")
        End Using
    End Sub
End Class
```

```

        Console.WriteLine()
        ' Split in two parts:
        ' =====

        splitter.Split(inputFile, "part1.pdf", "part2.pdf", 3)
        ' (!) Note: page number is 1-based.
        Console.WriteLine("Splitted at page 3 to files "part1.pdf"
and "part2.pdf"")
        Console.WriteLine()
        ' Split by ranges:
        ' =====

        Dim files As String() = splitter.Split(inputFile, "1-3,4-
6,7,8-")
        ' (!) Note: page numbers are 1-based; ending "-" means "to
the end".

        Console.WriteLine("Splitted by ranges: ")
        For Each file As String In files
            Console.WriteLine("    " & Path.GetFileName(file))
        Next
    End Using

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

End Sub
End Class

```

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout PDF Extractor SDK](#)

[Explore documentation](#)

[Visit www.ByteScout.com](#)

or

[Get Your Free API Key for www.PDF.co Web API](#)