

## How to find table in pdf and extract as xml with pdf extractor sdk in VB.NET and ByteScout PDF Suite

Learn to find table in pdf and extract as xml with pdf extractor sdk in VB.NET

An easy to understand sample source code to learn how to find table in pdf and extract as xml with pdf extractor sdk in VB.NET ByteScout PDF Suite: the set that includes 6 SDK products to work with PDF from generating rich PDF reports to extracting data from PDF documents and converting them to HTML. This bundle includes PDF (Generator) SDK, PDF Renderer SDK, PDF Extractor SDK, PDF to HTML SDK, PDF Viewer SDK and PDF Generator SDK for Javascript. It can find table in pdf and extract as xml with pdf extractor sdk in VB.NET.

The SDK samples given below describe how to quickly make your application do find table in pdf and extract as xml with pdf extractor sdk in VB.NET with the help of ByteScout PDF Suite. Just copy and paste the code into your VB.NET application's code and follow the instructions. Further improvement of the code will make it more robust.

All these programming tutorials along with source code samples and ByteScout free trial version are available for download from our website.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout PDF Suite](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout PDF Suite](#)

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

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

Source Code Files:

```
Imports Bytescout.PDFExtractor

Class Program
    Friend Shared Sub Main(args As String())

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

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

        ' We should define what kind of tables we should detect.
        ' So we set min required number of columns to 3 ...
        tableDetector.DetectionMinNumberOfColumns = 3
        ' ... and we set min required number of rows to 3
        tableDetector.DetectionMinNumberOfRows = 3

        ' Load sample PDF document
        xmlExtractor.LoadDocumentFromFile(".\sample3.pdf")
        tableDetector.LoadDocumentFromFile(".\sample3.pdf")

        ' Get page count
        Dim pageCount As Integer = tableDetector.GetPageCount()

        For i As Integer = 0 To pageCount - 1
            Dim t As Integer = 1
            ' Find first table and continue if found
            If (tableDetector.FindTable(i)) Then
                Do
                    ' Set extraction area for XML extractor to rectangle received from
                    xmlExtractor.SetExtractionArea(tableDetector.FoundTableLocation)
                    ' Export the table to XML file
                    xmlExtractor.SavePageXMLToFile(i, "page-" + i.ToString() + "-table-
                    t = t + 1
                Loop While tableDetector.FindNextTable()
            End If
        Next

        ' Cleanup
        xmlExtractor.Dispose()
        tableDetector.Dispose()

        ' Open first output file in default associated application (for demo purposes)
        System.Diagnostics.Process.Start("page-0-table-1.xml")

    End Sub
End Class
```

---

## VIDEO

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

## ON-PREMISE OFFLINE SDK

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