

How to convert protected pdf document to xls with pdf extractor sdk in VB.NET with ByteScout Premium Suite

Learning is essential in computer world and the tutorial below will demonstrate how to convert protected pdf document to xls with pdf extractor sdk in VB.NET

The sample source code below will teach you how to convert protected pdf document to xls with pdf extractor sdk in VB.NET. ByteScout Premium Suite is the bundle that includes twelve SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording. It can be applied to convert protected pdf document to xls with pdf extractor sdk using VB.NET.

The following code snippet for ByteScout Premium Suite works best when you need to quickly convert protected pdf document to xls with pdf extractor sdk in your VB.NET application. Follow the instructions from scratch to work and copy the VB.NET code. If you want to use these VB.NET sample examples in one or many applications then they can be used easily.

You can download free trial version of ByteScout Premium Suite from our website with this and 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 Bytescout.PDFExtractor
Imports System.Diagnostics

Class Program

    Friend Shared Sub Main(args As String())

        ' Create Bytescout.PDFExtractor.XLSExtractor instance

        Dim extractor As New XLSExtractor()
        extractor.RegistrationName = "demo"
        extractor.RegistrationKey = "demo"

        File.Delete("output.xls")

        ''' Document Password Can be set in two ways
        ''' 1. Using Property
        ''' 2. Using Events

        ' 1. Handle document password using Property
        ' extractor.Password = "password"

        ' 2. Handle document password using Event
        AddHandler extractor.PasswordRequired, New System.EventHandler(AddressOf extractor.PasswordRequired)

        ' Load sample PDF document
        extractor.LoadDocumentFromFile("encrypted (password is 'password').pdf")

        ' Uncomment this line if you need all pages converted into a single worksheet:
        ' extractor.PageToWorksheet = False

        ' Set the output format to XLS
        extractor.OutputFormat = SpreadsheetOutputFormat.XLS

        ' Save the spreadsheet to file
        extractor.SaveToXLSFile("output.xls")

        ' Cleanup
        extractor.Dispose()

        ' Open result file in default associated application (for demo purposes)
        Process.Start("output.xls")

    End Sub

    Private Shared Sub extractor_PasswordRequired(sender As Object, e As EventArgs)

        ' Ask user for password and put it to `Password` property.
        CType(sender, XLSExtractor).Password = "password"

    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