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

How to write a robust code in VB.NET to convert protected pdf document to xls with pdf extractor sdk with this step-by-step tutorial

Every ByteScout tool includes simple example VB.NET source codes that you can get here or in the folder with installed ByteScout product. ByteScout PDF Suite is 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 be applied to convert protected pdf document to xls with pdf extractor sdk using VB.NET.

The following code snippet for ByteScout PDF Suite works best when you need to quickly convert protected pdf document to xls with pdf extractor sdk in your VB.NET application. Simply copy and paste in your VB.NET project or application you and then run your app! This basic programming language sample code for VB.NET will do the whole work for you to convert protected pdf document to xls with pdf extractor sdk.

Trial version of ByteScout PDF Suite is available for free. Source code samples are included to help you with your VB.NET app.

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

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 extra
        ' 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 = SpreadseetOutputFormat.XLS
        ' Save the spreadsheet to file
        extractor.SaveToXLSFile("output.xls")
        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
```

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

visit www.PDF.co

www.bytescout.com