How to convert PDF to text from URL asynchronously for PDF to text API in VB.NET and ByteScout Cloud API Server

Follow this simple tutorial to learn convert PDF to text from URL asynchronously to have PDF to text API in VB.NET

Every ByteScout tool includes sample VB.NET source codes that you can find here or in the folder with installed ByteScout product. ByteScout Cloud API Server was designed to assist PDF to text API in VB.NET. ByteScout Cloud API Server is the ready to use Web API Server that can be deployed in less than 30 minutes into your own in-house server or into private cloud server. Can store data on in-house local server based storage or in Amazon AWS S3 bucket. Processing data solely on the server using buil-in ByteScout powered engine, no cloud services are used to process your data!.

Want to learn quickly? These fast application programming interfaces of ByteScout Cloud API Server for VB.NET plus the instruction and the code below will help to learn how to convert PDF to text from URL asynchronously. Follow the tutorial and copy - paste code for VB.NET into your project's code editor. You can use these VB.NET sample examples in one or many applications.

ByteScout Cloud API Server - free trial version is available on our website. Also, there are other code samples to help you with your VB.NET application included into trial version.

FOR MORE INFORMATION AND FREE TRIAL:

Download Free Trial SDK (on-premise version)

Read more about ByteScout Cloud API Server

Explore API Documentation

Get Free Training for ByteScout Cloud API Server

Get Free API key for Web API

visit www.ByteScout.com

Source Code Files:

ByteScoutWebApiExample.sln

```
Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio 15
VisualStudioVersion = 15.0.26730.10
MinimumVisualStudioVersion = 10.0.40219.1
Project("{F184B08F-C81C-45F6-A57F-5ABD9991F28F}") = "ByteScoutWebApiExample", "ByteScou
EndProject
Global
        GlobalSection(SolutionConfigurationPlatforms) = preSolution
                Debug|Any CPU = Debug|Any CPU
                Release|Any CPU = Release|Any CPU
        EndGlobalSection
        GlobalSection(ProjectConfigurationPlatforms) = postSolution
                {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Debug|Any CPU.ActiveCfg = Debug
                {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Debug|Any CPU.Build.0 = Debug|An
                {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Release|Any CPU.ActiveCfg = Rele
                {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Release|Any CPU.Build.0 = Releas
        EndGlobalSection
        GlobalSection(SolutionProperties) = preSolution
                HideSolutionNode = FALSE
        EndGlobalSection
        GlobalSection(ExtensibilityGlobals) = postSolution
                SolutionGuid = {4576C9BB-A42D-46A8-9198-7E2982E122FA}
        EndGlobalSection
EndGlobal
```

Module1.vb

```
Imports System.IO
Imports System.Net
Imports System.Threading
Imports Newtonsoft.Json.Linq

' Please NOTE: In this sample we're assuming Cloud Api Server is hosted at "https://lo
' If it's not then please replace this with with your hosting url.

' Cloud API asynchronous "PDF To Text" job example.
' Allows to avoid timeout errors when processing huge or scanned PDF documents.

Module Module1

        ' Direct URL of source PDF file.
        Const SourceFileUrl As String = "https://bytescout-com.s3.amazonaws.com/files/
        ' Comma-separated list of page indices (or ranges) to process. Leave empty for
        Const Pages As String = ""
        ' PDF document password. Leave empty for unprotected documents.
        Const Password As String = ""
        ' Destination TXT file name
        Const DestinationFile As String = ".\result.txt"
        ' (!) Make asynchronous job
        Const Async As Boolean = True

        Sub Main()

                ' Create standard .NET web client instance
                Dim webClient As WebClient = New WebClient()

                ' Prepare URL for `PDF To TXT` API call
                Dim query As String = Uri.EscapeUriString(String.Format(
                        "https://localhost/pdf/convert/to/text?name={0}&password={1}&pa
                        Path.GetFileName(DestinationFile),
                        Password,
                        Pages,
                        SourceFileUrl,
                        Async))

                Try
                        ' Execute request
                        Dim response As String = webClient.DownloadString(query)

                        ' Parse JSON response
                        Dim json As JObject = JObject.Parse(response)

                        If json("error").ToObject(Of Boolean) = False Then

                                ' Asynchronous job ID
                                Dim jobId As String = json("jobId").ToString()
                                ' URL of generated TXT file that will available after t
                                Dim resultFileUrl As String = json("url").ToString()

                                ' Check the job status in a loop.
                                ' If you don't want to pause the main thread you can re
                                ' to use a separate thread for the status checking and
                                Do
                                        Dim status As String = CheckJobStatus(jobId) '

                                        ' Display timestamp and status (for demo purpos
                                        Console.WriteLine(DateTime.Now.ToLongTimeString

                                        If status = "success" Then

                                                ' Download TXT file
                                                webClient.DownloadFile(resultFileUrl, D

                                                Console.WriteLine("Generated TXT file s
                                                Exit Do

                                        ElseIf status = "working" Then

                                                ' Pause for a few seconds
                                                Thread.Sleep(3000)

                                        Else
```

```
                                                Console.WriteLine(status)
                                                Exit Do

                                        End If

                                Loop

                        Else
                                Console.WriteLine(json("message").ToString())
                        End If

                Catch ex As WebException
                        Console.WriteLine(ex.ToString())
                End Try

                webClient.Dispose()

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

        End Sub

        Function CheckJobStatus(jobId As String) As String

                Using webClient As WebClient = New WebClient()

                        Dim url As String = "https://localhost/job/check?jobid=" + jobI

                        Dim response As String = webClient.DownloadString(url)
                        Dim json As JObject = JObject.Parse(response)

                        Return Convert.ToString(json("status"))

                End Using

        End Function

End Module
```

packages.config

```
<?xml version="1.0" encoding="utf-8"?>
<packages>
  <package id="Newtonsoft.Json" version="10.0.3" targetFramework="net40" />
</packages>
```

VIDEO

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

ON-PREMISE OFFLINE SDK

60 Day Free Trial or Visit ByteScout Cloud API Server Home Page
Explore ByteScout Cloud API Server Documentation
Explore Samples
Sign Up for ByteScout Cloud API Server 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