

## PDF to CSV API in VB.NET and PDF.co Web API

### Build PDF to CSV API in VB.NET

:

### Step-by-step instructions on how to do PDF to CSV API in VB.NET

Here you may get thousands pre-made source code samples for simple implementation in your own programming VB.NET projects. PDF.co Web API was created to assist PDF to CSV API in VB.NET. PDF.co Web API is the Rest API that provides set of data extraction functions, tools for documents manipulation, splitting and merging of pdf files. Includes built-in OCR, images recognition, can generate and read barcodes from images, scans and pdf.

If you want to quickly learn then these fast application programming interfaces of PDF.co Web API for VB.NET plus the guideline and the VB.NET code below will help you quickly learn PDF to CSV API. If you want to know how it works, then this VB.NET sample code should be copied and pasted into your application's code editor. Then just compile and run it. Want to see how it works with your data then code testing will allow the function to be tested and work properly.

PDF.co Web API is available as a free trial. You may get it from our website along with all other source code samples for VB.NET applications.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about PDF.co Web API](#)

[Explore API Documentation](#)

[Get Free Training for PDF.co Web API](#)

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

[visit www.ByteScout.com](http://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", "ByteScoutWebApiExample
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|Any CPU
        {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Debug|Any CPU.Build.0 = Debug|Any CPU
        {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Release|Any CPU.ActiveCfg = Release|Any CPU
        {9B91124C-66C3-4BD9-B29E-168C1ABB15AC}.Release|Any CPU.Build.0 = Release|Any CPU
    EndGlobalSection
    GlobalSection(SolutionProperties) = preSolution
        HideSolutionNode = FALSE
    EndGlobalSection
    GlobalSection(ExtensibilityGlobals) = postSolution
        SolutionGuid = {4576C9BB-A42D-46A8-9198-7E2982E122FA}
    EndGlobalSection
EndGlobal
```

## ByteScoutWebApiExample.vbproj

```
<?xml version="1.0" encoding="utf-8"?>
<Project ToolsVersion="15.0" xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
  <Import Project="$(MSBuildExtensionsPath)\$(MSBuildToolsVersion)\Microsoft.Common.props" Condition="Exists('
  <PropertyGroup>
    <Configuration Condition="'$(Configuration)' == "">Debug</Configuration>
    <Platform Condition="'$(Platform)' == "">AnyCPU</Platform>
    <ProjectGuid>{9B91124C-66C3-4BD9-B29E-168C1ABB15AC}</ProjectGuid>
    <OutputType>Exe</OutputType>
    <StartupObject>ByteScoutWebApiExample.Module1</StartupObject>
    <RootNamespace>ByteScoutWebApiExample</RootNamespace>
    <AssemblyName>ByteScoutWebApiExample</AssemblyName>
    <FileAlignment>512</FileAlignment>
    <MyType>Console</MyType>
    <TargetFrameworkVersion>v4.0</TargetFrameworkVersion>
  </PropertyGroup>
  <PropertyGroup Condition="'$(Configuration)|$(Platform)' == 'Debug|AnyCPU' ">
    <PlatformTarget>AnyCPU</PlatformTarget>
    <DebugSymbols>>true</DebugSymbols>
    <DebugType>full</DebugType>
    <DefineDebug>true</DefineDebug>
    <DefineTrace>true</DefineTrace>
    <OutputPath>bin\Debug</OutputPath>
    <DocumentationFile>ByteScoutWebApiExample.xml</DocumentationFile>
    <NoWarn>42016,41999,42017,42018,42019,42032,42036,42020,42021,42022</NoWarn>
  </PropertyGroup>
  <PropertyGroup Condition="'$(Configuration)|$(Platform)' == 'Release|AnyCPU' ">
    <PlatformTarget>AnyCPU</PlatformTarget>
    <DebugType>pdbonly</DebugType>
    <DefineDebug>>false</DefineDebug>
```

```
<DefineTrace>>true</DefineTrace>
<Optimize>>true</Optimize>
<OutputPath>bin\Release\</OutputPath>
<DocumentationFile>ByteScoutWebApiExample.xml</DocumentationFile>
<NoWarn>42016,41999,42017,42018,42019,42032,42036,42020,42021,42022</NoWarn>
</PropertyGroup>
<PropertyGroup>
  <OptionExplicit>On</OptionExplicit>
</PropertyGroup>
<PropertyGroup>
  <OptionCompare>Binary</OptionCompare>
</PropertyGroup>
<PropertyGroup>
  <OptionStrict>Off</OptionStrict>
</PropertyGroup>
<PropertyGroup>
  <OptionInfer>On</OptionInfer>
</PropertyGroup>
<ItemGroup>
  <Reference Include="Newtonsoft.Json, Version=10.0.0.0, Culture=neutral, PublicKeyToken=30ad4fe6b2a6aeed, p
    <HintPath>packages\Newtonsoft.Json.10.0.3\lib\net40\Newtonsoft.Json.dll</HintPath>
  </Reference>
  <Reference Include="System" />
  <Reference Include="System.Data" />
  <Reference Include="System.Deployment" />
  <Reference Include="System.Xml" />
  <Reference Include="System.Core" />
  <Reference Include="System.Xml.Linq" />
  <Reference Include="System.Data.DataSetExtensions" />
</ItemGroup>
<ItemGroup>
  <Import Include="Microsoft.VisualBasic" />
  <Import Include="System" />
  <Import Include="System.Collections" />
  <Import Include="System.Collections.Generic" />
  <Import Include="System.Data" />
  <Import Include="System.Diagnostics" />
  <Import Include="System.Linq" />
  <Import Include="System.Xml.Linq" />
</ItemGroup>
<ItemGroup>
  <Compile Include="Module1.vb" />
  <Compile Include="My Project\AssemblyInfo.vb" />
  <Compile Include="My Project\Application.Designer.vb">
    <AutoGen>True</AutoGen>
    <DependentUpon>Application.myapp</DependentUpon>
  </Compile>
  <Compile Include="My Project\Resources.Designer.vb">
    <AutoGen>True</AutoGen>
    <DesignTime>True</DesignTime>
    <DependentUpon>Resources.resx</DependentUpon>
  </Compile>
  <Compile Include="My Project\Settings.Designer.vb">
    <AutoGen>True</AutoGen>
    <DependentUpon>Settings.settings</DependentUpon>
    <DesignTimeSharedInput>True</DesignTimeSharedInput>
  </Compile>
</ItemGroup>
<ItemGroup>
  <EmbeddedResource Include="My Project\Resources.resx">
    <Generator>VbMyResourcesResXFileCodeGenerator</Generator>
    <LastGenOutput>Resources.Designer.vb</LastGenOutput>
    <CustomToolNamespace>My.Resources</CustomToolNamespace>
    <SubType>Designer</SubType>
  </EmbeddedResource>
</ItemGroup>
<ItemGroup>
  <None Include="My Project\Application.myapp">
    <Generator>MyApplicationCodeGenerator</Generator>
    <LastGenOutput>Application.Designer.vb</LastGenOutput>
  </None>
  <None Include="My Project\Settings.settings">
    <Generator>SettingsSingleFileGenerator</Generator>
    <CustomToolNamespace>My</CustomToolNamespace>
    <LastGenOutput>Settings.Designer.vb</LastGenOutput>
  </None>
  <None Include="packages.config" />
  <Content Include="sample-rotated.pdf">
```

```

    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </Content>
  <Content Include="profile.json">
    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </Content>
</ItemGroup>
<Import Project="$(MSBuildToolsPath)\Microsoft.VisualBasic.targets" />
</Project>

```

## Module1.vb

```

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

Module Module1

    ' The authentication key (API Key).
    ' Get your own by registering at https://app.pdf.co/documentation/api
    Const API_KEY As String = "*****"

    ' Source PDF file
    Const SourceFile As String = ".\sample-rotated.pdf"
    ' Comma-separated list of page indices (or ranges) to process. Leave empty for all pages. Example: '0,2-5,7-'
    Const Pages As String = ""
    ' PDF document password. Leave empty for unprotected documents.
    Const Password As String = ""
    ' Destination CSV file name
    Const DestinationFile As String = ".\result.csv"

    ' Sample profile that sets advanced conversion options.
    ' Advanced options are properties of CSVExtractor class from ByteScout PDF Extractor SDK used in the back-end:
    ' https://cdn.bytescout.com/help/BytescoutPDFExtractorSDK/html/87ce5fa6-3143-167d-abbd-bc7b5e160fe5.htm

    'Valid RotationAngle values
    '0 - no rotation
    '1 - 90 degrees
    '2 - 180 degrees
    '3 - 270 degrees
    ReadOnly Profiles As String = File.ReadAllText("profile.json")

    Sub Main()

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

        ' Set API Key
        webClient.Headers.Add("x-api-key", API_KEY)

        ' 1. RETRIEVE THE PRESIGNED URL TO UPLOAD THE FILE.
        ' * If you already have a direct file URL, skip to the step 3.

        ' Prepare URL for `Get Presigned URL` API call
        Dim query As String = Uri.EscapeUriString(String.Format(
            "https://api.pdf.co/v1/file/upload/get-presigned-url?contenttype=application/octet-stream&name={0}",
            Path.GetFileName(SourceFile)))

        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
                ' Get URL to use for the file upload
                Dim uploadUrl As String = json("presignedUrl").ToString()
            }
        }
    }

```

```

' Get URL of uploaded file to use with later API calls
Dim uploadedFileUrl As String = json("url").ToString()

' 2. UPLOAD THE FILE TO CLOUD.

webClient.Headers.Add("content-type", "application/octet-stream")
webClient.UploadFile(uploadUrl, "PUT", SourceFile) ' You can use UploadData() instead if your file is byte a

' 3. CONVERT UPLOADED PDF FILE TO CSV

' Prepare URL for `PDF To CSV` API call
query = Uri.EscapeUriString(String.Format(
    "https://api.pdf.co/v1/pdf/convert/to/csv?name={0}&password={1}&pages={2}&url={3}&profiles={4}",
    Path.GetFileName(DestinationFile),
    Password,
    Pages,
    uploadedFileUrl,
    Profiles))

' Execute request
response = webClient.DownloadString(query)

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

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

    ' Get URL of generated CSV file
    Dim resultFileUrl As String = json("url").ToString()

    ' Download CSV file
    webClient.DownloadFile(resultFileUrl, DestinationFile)

    Console.WriteLine("Generated CSV file saved as ""{0}"" file.", DestinationFile)

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

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

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 PDF.co Web API Home Page](#)  
[Explore PDF.co Web API Documentation](#)  
[Explore Samples](#)  
[Sign Up for PDF.co Web API 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)