www.bytescout.com

How to convert PDF to XLSX from URL asynchronously for PDF to excel API in C# and PDF.co Web API

See how to convert PDF to XLSX from URL asynchronously to have PDF to excel API in C#

Today you are going to learn how to convert PDF to XLSX from URL asynchronously in C#. PDF.co Web API was made to help with PDF to excel API in C#. PDF.co Web API is the Web API with a set of tools for documents manipulation, data conversion, data extraction, splitting and merging of documents. Includes image recognition, built-in OCR, barcode generation and barcode decoders to decode bar codes from scans, pictures and pdf.

C# code samples for C# developers help to speed up the application's code writing when using PDF.co Web API. For implimentation of this functionality, please copy and paste code below into your app using code editor. Then compile and run your app. Enjoy writing a code with ready-to-use sample C# codes to add PDF to excel API functions using PDF.co Web API in C#.

ByteScout free trial version is available for FREE download from our website. Programming tutorials along with source code samples are included.

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

Source Code Files:

ByteScoutWebApiExample.sln

```
Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio 2013
VisualStudioVersion = 12.0.40629.0
MinimumVisualStudioVersion = 10.0.40219.1
Project("{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}") = "ByteScoutWebApiExample", "ByteScou
EndProject
Global
        GlobalSection(SolutionConfigurationPlatforms) = preSolution
                Debug|Any CPU = Debug|Any CPU
                Release|Any CPU = Release|Any CPU
        EndGlobalSection
        GlobalSection(ProjectConfigurationPlatforms) = postSolution
                {1E1C2C34-017E-4605-AE2B-55EA3313BE51}.Debug|Any CPU.ActiveCfg = Debug
                {1E1C2C34-017E-4605-AE2B-55EA3313BE51}.Debug|Any CPU.Build.0 = Debug|A
                {1E1C2C34-017E-4605-AE2B-55EA3313BE51}.Release|Any CPU.ActiveCfg = Rel
                {1E1C2C34-017E-4605-AE2B-55EA3313BE51}.Release|Any CPU.Build.0 = Releas
        EndGlobalSection
        GlobalSection(SolutionProperties) = preSolution
                HideSolutionNode = FALSE
        EndGlobalSection
EndGlobal
```

Program.cs

```
using System;
using System.IO;
using System.Net;
using Newtonsoft.Json.Linq;
using System.Threading;


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

namespace ByteScoutWebApiExample
{
        class Program
        {
                // The authentication key (API Key).
                // Get your own by registering at https://app.pdf.co/documentation/api
                const String API_KEY = "**********************************";

                // Direct URL of source PDF file.
                const string SourceFileUrl = "https://bytescout-com.s3.amazonaws.com/f
                // Comma-separated list of page indices (or ranges) to process. Leave
```

```
const string Pages = "";
// PDF document password. Leave empty for unprotected documents.
const string Password = "";
// Destination XLSX file name
const string DestinationFile = @".\result.xlsx";
// (!) Make asynchronous job
const bool Async = true;


static void Main(string[] args)
{
        // Create standard .NET web client instance
        WebClient webClient = new WebClient();

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

        // Prepare URL for `PDF To XLSX` API call
        string query = Uri.EscapeUriString(string.Format(
                "https://api.pdf.co/v1/pdf/convert/to/xlsx?name={0}&pas
                Path.GetFileName(DestinationFile),
                Password,
                Pages,
                SourceFileUrl,
                Async));

        try
        {
                // Execute request
                string response = webClient.DownloadString(query);

                // Parse JSON response
                JObject json = JObject.Parse(response);

                if (json["error"].ToObject<bool>() == false)
                {
                        // Asynchronous job ID
                        string jobId = json["jobId"].ToString();
                        // URL of generated XLSX file that will availab
                        string resultFileUrl = json["url"].ToString();

                        // Check the job status in a loop.
                        // If you don't want to pause the main thread y
                        // to use a separate thread for the status chec
                        do
                        {
                                string status = CheckJobStatus(jobId);

                                // Display timestamp and status (for de
                                Console.WriteLine(DateTime.Now.ToLongT

                                if (status == "success")
                                {
                                        // Download XLSX file
                                        webClient.DownloadFile(resultF

                                        Console.WriteLine("Generated XL
                                        break;
                                }
                                else if (status == "working")
```

```
                                        {
                                                // Pause for a few seconds
                                                Thread.Sleep(3000);
                                        }
                                        else
                                        {
                                                Console.WriteLine(status);
                                                break;
                                        }
                                }
                                while (true);
                        }
                        else
                        {
                                Console.WriteLine(json["message"].ToString());
                        }
                }
                catch (WebException e)
                {
                        Console.WriteLine(e.ToString());
                }

                webClient.Dispose();


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

        static string CheckJobStatus(string jobId)
        {
                using (WebClient webClient = new WebClient())
                {
                        // Set API Key
        webClient.Headers.Add("x-api-key", API_KEY);

                        string url = "https://api.pdf.co/v1/job/check?jobid=" +

                        string response = webClient.DownloadString(url);
                        JObject json = JObject.Parse(response);

                        return Convert.ToString(json["status"]);
                }
        }
    }
}
```

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

visit www.PDF.co