How to PDF text search API in C# with PDF.co Web API

Write code in C# to PDF text search API with this step-by-step tutorial

Source code documentation samples provide quick and easy way to add a required functionality into your application. PDF.co Web API can PDF text search API. It can be used from C#. PDF.co Web API is the flexible Web API that includes full set of functions from e-signature requests to data extraction, OCR, images recognition, pdf splitting and pdf splitting. Can also generate barcodes and read barcodes from images, scans and pdf.

You will save a lot of time on writing and testing code as you may just take the C# code from PDF.co Web API for PDF text search API below and use it in your application. Just copy and paste the code into your C# application's code and follow the instruction. Detailed tutorials and documentation are available along with installed PDF.co Web API if you'd like to dive deeper into the topic and the details of the API.

ByteScout free trial version is available for download from our website. It includes all these programming tutorials along with source code samples.

C# - Program.cs

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

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 = "*****************************************";

        // Source PDF file
        const string SourceFile = @".\sample.pdf";

        // Comma-separated list of page indices (or ranges) to process. Leave empty
for all pages. Example: '0,2-5,7-'.
        const string Pages = "";

        // PDF document password. Leave empty for unprotected documents.
        const string Password = "";

        // Search string.
```

```
        const string SearchString = @"\d{1,}\.\d\d"; // Regular expression to find
numbers like '100.00'
                                                     // Note: do not use `+` char in
regex, but use `{1,}` instead.
                                                     // `+` char is valid for URL and
will not be escaped, and it will become a space char on the server side.

        // Enable regular expressions (Regex)
        const bool RegexSearch = true;

        // (!) 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);

            // 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
            string query = 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
                string response = webClient.DownloadString(query);

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

                if (json["error"].ToObject() == false)
                {
                    // Get URL to use for the file upload
                    string uploadUrl = json["presignedUrl"].ToString();
                    string uploadedFileUrl = 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[] or Stream

                    // 3. MAKE UPLOADED PDF FILE SEARCHABLE

                    // Prepare URL for `PDF Text Search` API call
                    // See documentation:
https://app.pdf.co/documentation/api/1.0/pdf/find.html
                    query = Uri.EscapeUriString(string.Format(
                                "https://api.pdf.co/v1/pdf/find?password={0}&pages=
{1}&url={2}&searchString={3}®exSearch={4}&async={5}",
                                Password,
                                Pages,
```

```
                                uploadedFileUrl,
                                SearchString,
                                RegexSearch,
                                Async));

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

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

                    if (json["error"].ToObject() == false)
                    {
                        // Asynchronous job ID
                        string jobId = json["jobId"].ToString();

                        // URL of generated json file that will available after the
job completion
                        string resultFileUrl = json["url"].ToString();

                        // Check the job status in a loop.
                        // If you don't want to pause the main thread you can rework
the code
                        // to use a separate thread for the status checking and
completion.
                        do
                        {
                            string status = CheckJobStatus(jobId); // Possible
statuses: "working", "failed", "aborted", "success".

                            // Display timestamp and status (for demo purposes)
                            Console.WriteLine(DateTime.Now.ToLongTimeString() + ": "
+ status);

                            if (status == "success")
                            {
                                // Execute request
                                string respFileJson =
webClient.DownloadString(resultFileUrl);

                                // Parse JSON response
                                JArray jsonFoundInformation =
JArray.Parse(respFileJson);

                                // Display found information in console
                                foreach (JToken item in jsonFoundInformation)
                                {
                                    Console.WriteLine($"Found text \"{item["text"]}\"
at coordinates {item["left"]}, {item["top"]}");
                                }

                                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());
                }
            }
            else
            {
                Console.WriteLine(json["message"].ToString());
            }
        }
        catch (WebException ex)
        {
            Console.WriteLine(ex.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=" + jobId;

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

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

}
```

C# - packages.config

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about PDF.co Web API]

[Explore documentation]

[Visit www.ByteScout.com]

or

[Get Your Free API Key for www.PDF.co Web API]