www.bytescout.com

# How to convert PDF to CSV from uploaded file (node for PDF to CSV API in JavaScript using ByteScout Cloud API Server

## Learn to write code convert PDF to CSV from uploaded file (node for PDF to CSV API in JavaScript: Simple How To Tutorial

These source code samples are listed and grouped by their programming language and functions they use. ByteScout Cloud API Server was designed to assist PDF to CSV API in JavaScript. ByteScout Cloud API Server is the ready to deploy Web API Server that can be deployed in less than thirty minutes into your own in-house Windows server (no Internet connnection is required to process data!) 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 built-in ByteScout powered engine, no cloud services are used to process your data!.

JavaScript code snippet like this for ByteScout Cloud API Server works best when you need to quickly implement PDF to CSV API in your JavaScript application. Follow the tutorial and copy - paste code for JavaScript into your project's code editor. You can use these JavaScript 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 JavaScript 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:

ConvertPdfToCsvFromUploadedFile.js

```
/*jshint esversion: 6 */

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

var https = require("https");
var path = require("path");
var fs = require("fs");

// `request` module is required for file upload.
// Use "npm install request" command to install.
var request = require("request");
 // Source PDF file
const SourceFile = "./sample.pdf";
// Comma-separated list of page indices (or ranges) to process. Leave empty for all pa
const Pages = "";
// PDF document password. Leave empty for unprotected documents.
const Password = "";
// Destination CSV file name
const DestinationFile = "./result.csv";

// 1. RETRIEVE PRESIGNED URL TO UPLOAD FILE.
getPresignedUrl(SourceFile)
.then(([uploadUrl, uploadedFileUrl]) => {
    // 2. UPLOAD THE FILE TO CLOUD.
    uploadFile(SourceFile, uploadUrl)
    .then(() => {
        // 3. CONVERT UPLOADED PDF FILE TO CSV
        convertPdfToCsv(uploadedFileUrl, Password, Pages, DestinationFile);
    })
    .catch(e => {
        console.log(e);
    });
})
.catch(e => {
    console.log(e);
});


function getPresignedUrl(localFile) {
    return new Promise(resolve => {
        // Prepare request to `Get Presigned URL` API endpoint
        let queryPath = `/file/upload/get-presigned-url?name=${path.basename(SourceFil
        let reqOptions = {
            host: "localhost",
            path: encodeURI(queryPath)
        };
        // Send request
        https.get(reqOptions, (response) => {
            response.on("data", (d) => {
                let data = JSON.parse(d);
                if (data.error == false) {
                    // Return presigned url we received
```

```
                    resolve([data.presignedUrl, data.url]);
                }
                else {
                    // Service reported error
                    console.log("getPresignedUrl(): " + data.message);
                }
            });
        })
        .on("error", (e) => {
            // Request error
            console.log("getPresignedUrl(): " + e);
        });
    });
}

function uploadFile(localFile, uploadUrl) {
    return new Promise(resolve => {
        fs.readFile(SourceFile, (err, data) => {
            request({
                method: "PUT",
                url: uploadUrl,
                body: data,
            }, (err, res, body) => {
                if (!err) {
                    resolve();
                }
                else {
                    console.log("uploadFile() request error: " + e);
                }
            });
        });
    });
}

function convertPdfToCsv(uploadedFileUrl, password, pages, destinationFile) {
    // Prepare request to `PDF To CSV` API endpoint
    var queryPath = `/pdf/convert/to/csv?name=${path.basename(destinationFile)}&passwo
    let reqOptions = {
        host: "localhost",
        path: encodeURI(queryPath),
        method: "GET"
    };
    // Send request
    https.get(reqOptions, (response) => {
        response.on("data", (d) => {
            response.setEncoding("utf8");

            // Parse JSON response
            let data = JSON.parse(d);
            console.log(`Job #${data.jobId} has been created!`);

            if (data.error == false) {
                checkIfJobIsCompleted(data.jobId, data.url, destinationFile);
            }
            else {
                // Service reported error
                console.log("convertPdfToCsv(): " + data.message);
            }
        });
    })
```

```
    .on("error", (e) => {
        // Request error
        console.log("convertPdfToCsv(): " + e);
    });
}

function checkIfJobIsCompleted(jobId, resultFileUrl, destinationFile) {
    let queryPath = `/job/check?jobid=${jobId}`;
    let reqOptions = {
        host: "localhost",
        path: encodeURI(queryPath),
        method: "GET"
    };

    https.get(reqOptions, (response) => {
        response.on("data", (d) => {
            response.setEncoding("utf8");

            // Parse JSON response
            let data = JSON.parse(d);
            console.log(`Checking Job #${jobId}, Status: ${data.status}, Time: ${new Da

            if (data.status == "working") {
                // Check again after 3 seconds
                setTimeout(function(){ checkIfJobIsCompleted(jobId, resultFileUrl, dest
            }
            else if (data.status == "success") {
                // Download CSV file
                var file = fs.createWriteStream(destinationFile);
                https.get(resultFileUrl, (response2) => {
                    response2.pipe(file)
                        .on("close", () => {
                            console.log(`Generated CSV file saved as "${destinationFile
                        });
                });
            }
            else {
                console.log(`Operation ended with status: "${data.status}".`);
            }
        })
    });
}
```

---

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