How to convert PDF to JPEG from uploaded file (node for PDF to image API in JavaScript using PDF.co Web API

See how to convert PDF to JPEG from uploaded file (node to have PDF to image API in JavaScript

The sample source codes on this page will demonstrate you how to make PDF to image API in JavaScript. PDF to image API in JavaScript can be implemented with PDF.co Web API. 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.

This rich sample source code in JavaScript for PDF.co Web API includes the number of functions and options you should do calling the API to implement PDF to image API. Open your JavaScript project and simply copy & paste the code and then run your app! Tutorials 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.

Trial version of ByteScout is available for free download from our website. This and other source code samples for JavaScript and other programming languages are available.

JavaScript - ConvertPdfToJpegFromUploadedFile.js

```
/*jshint esversion: 6 */

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");

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

// Source PDF file
const SourceFile = "./sample.pdf";
// Comma-separated list of page indices (or ranges) to process. Leave empty for all
pages. Example: '0,2-5,7-'.
const Pages = "";
// PDF document password. Leave empty for unprotected documents.
const Password = "";
```

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


function getPresignedUrl(apiKey, localFile) {
    return new Promise(resolve => {
        // Prepare request to `Get Presigned URL` API endpoint
        let queryPath = `/v1/file/upload/get-presigned-url?
contenttype=application/octet-stream&name=${path.basename(SourceFile)}`;
        let reqOptions = {
            host: "api.pdf.co",
            path: encodeURI(queryPath),
            headers: { "x-api-key": API_KEY }
        };
        // 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(apiKey, localFile, uploadUrl) {
    return new Promise(resolve => {
        fs.readFile(SourceFile, (err, data) => {
            request({
                method: "PUT",
                url: uploadUrl,
                body: data,
                headers: {
                    "Content-Type": "application/octet-stream"
                }
            }, (err, res, body) => {
```

```
                if (!err) {
                    resolve();
                }
                else {
                    console.log("uploadFile() request error: " + e);
                }
            });
        });
    });
}

function convertPdfToJpeg(apiKey, uploadedFileUrl, password, pages) {
    // Prepare URL for `PDF To JPEG` API call
    var queryPath = `/v1/pdf/convert/to/jpg?
password=${password}&pages=${pages}&url=${uploadedFileUrl}&async=True`;
    let reqOptions = {
        host: "api.pdf.co",
        path: encodeURI(queryPath),
        method: "GET",
        headers: { "x-api-key": API_KEY }
    };
    // Send request
    https.get(reqOptions, (response) => {
        response.on("data", (d) => {
            response.setEncoding("utf8");
            // Parse JSON response
            let data = JSON.parse(d);
            if (data.error == false) {
                console.log(`Job #${data.jobId} has been created!`);
                checkIfJobIsCompleted(data.jobId, data.url);
            }
            else {
                // Service reported error
                console.log("convertPdfToJpeg(): " + data.message);
            }
        });
    })
        .on("error", (e) => {
            // Request error
            console.log("convertPdfToJpeg(): " + e);
        });
}

function checkIfJobIsCompleted(jobId, resultFileUrlJson) {

    let queryPath = `/v1/job/check?jobid=${jobId}`;
    let reqOptions = {
        host: "api.pdf.co",
        path: encodeURI(queryPath),
        method: "GET",
        headers: { "x-api-key": API_KEY }
    };

    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
```

```
Date().toLocaleString()}`);

            if (data.status == "working") {
                // Check again after 3 seconds
                setTimeout(function(){checkIfJobIsCompleted(jobId,
resultFileUrlJson)} , 3000);
            }
            else if (data.status == "success") {

                request({ method: 'GET', uri: resultFileUrlJson, gzip: true },
                    function (error, response, body) {

                        // Parse JSON response
                        let respJsonFileArray = JSON.parse(body);
                        let page = 1;

                        respJsonFileArray.forEach((url) => {
                            var localFileName = `./page${page}.jpg`;
                            var file = fs.createWriteStream(localFileName);
                            https.get(url, (response2) => {
                                response2.pipe(file)
                                    .on("close", () => {
                                        console.log(`Generated JPEG file saved as
"${localFileName} file."`);
                                    });
                            });
                            page++;
                        }, this);

                    });
            }
            else {
                console.log(`Operation ended with status: "${data.status}".`);
            }
        })
    });
}
```

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]