

How to optimize PDF from uploaded file (node for PDF optimization API in JavaScript and PDF.co Web API

How to optimize PDF from uploaded file (node for PDF optimization API in JavaScript: How To Tutorial

Every ByteScout tool contains example JavaScript source codes that you can find here or in the folder with installed ByteScout product. PDF.co Web API helps with PDF optimization API in JavaScript. 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.

Fast application programming interfaces of PDF.co Web API for JavaScript plus the instruction and the code below will help to learn how to optimize PDF from uploaded file (node. Open your JavaScript project and simply copy & paste the code and then run your app! Test JavaScript sample code examples whether they respond your needs and requirements for the project.

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 - OptimizePdfFromUploadedFile.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";
// PDF document password. Leave empty for unprotected documents.
const Password = "";
// Destination PDF file name
const DestinationFile = "./result.pdf";
```

```

// 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. OPTIMIZE UPLOADED PDF FILE
    optimizePdf(API_KEY, uploadedFileUrl, Password, DestinationFile);
  })
  .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 optimizePdf(apiKey, uploadedFileUrl, password, destinationFile) {
    // Prepare request to `Optimize PDF` API endpoint
    var queryPath = `/v1/pdf/optimize?
name=${path.basename(destinationFile)}&password=${password}&url=${uploadedFileUrl}`;
    let reqOptions = {
        host: "api.pdf.co",
        path: encodeURI(queryPath),
        method: "GET",
        headers: { "x-api-key": API_KEY }
    };
    // Send request
    https.get(reqOptions, (response) => {
        response.on("data", (d) => {
            response.setEncoding("utf8");
            // Parse JSON response
            let data = JSON.parse(d);
            if (data.error == false) {
                // Download PDF file
                var file = fs.createWriteStream(destinationFile);
                https.get(data.url, (response2) => {
                    response2.pipe(file)
                    .on("close", () => {
                        console.log(`Generated PDF file saved as "${destinationFile}"
file.`);
                    });
                });
            }
            else {
                // Service reported error
                console.log("readBarcodes(): " + data.message);
            }
        });
    })
    .on("error", (e) => {
        // Request error
        console.log("readBarcodes(): " + e);
    });
}
}

```

FOR MORE INFORMATION AND FREE TRIAL:

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

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

[Explore documentation](#)

[Visit \[www.ByteScout.com\]\(http://www.ByteScout.com\)](#)

or

[Get Your Free API Key for \[www.PDF.co\]\(http://www.PDF.co\) Web API](#)