

## How to split PDF from uploaded file (node for PDF splitting API in JavaScript with ByteScout Cloud API Server

Learn to write code split PDF from uploaded file (node for PDF splitting API in JavaScript: Simple How To Tutorial

On this page, you will find sample source codes which show you how to handle a complex task, such as, PDF splitting API in JavaScript. ByteScout Cloud API Server was designed to assist PDF splitting 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 connection 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!.

Want to learn quickly? These fast application programming interfaces of ByteScout Cloud API Server for JavaScript plus the instruction and the code below will help to learn how to split PDF from uploaded file (node. Open your JavaScript project and simply copy & paste the code and then run your app! Use of ByteScout Cloud API Server in JavaScript is also described in the documentation given along with the product.

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.

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](http://www.ByteScout.com)

Source Code Files:

## SplitPdfFromUploadedFile.js

```
/*jshint esversion: 6 */

// Please NOTE: In this sample we're assuming Cloud Api Server is hosted at "https://1
// 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 to split
const SourceFile = "./sample.pdf";
// Comma-separated list of page numbers (or ranges) to process. Example: '1,3-5,7-'.
const Pages = "1-2,3-";

// 1. RETRIEVE PRESIGNED URL TO UPLOAD FILE.
getPresignedUrl(SourceFile)
.then(([uploadUrl, uploadedFileUrl]) => {
  // 2. UPLOAD THE FILE TO CLOUD.
  uploadFile(SourceFile, uploadUrl)
  .then(() => {
    // 3. SPLIT UPLOADED PDF
    splitPdf(uploadedFileUrl, Pages);
  })
  .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?contenttype=application/octet-s
    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,
                headers: {
                    "Content-Type": "application/octet-stream"
                }
            }, (err, res, body) => {
                if (!err) {
                    resolve();
                }
                else {
                    console.log("uploadFile() request error: " + e);
                }
            });
        });
    });
}

function splitPdf(uploadedFileUrl, pages) {
    // Prepare request to `Make Searchable PDF` API endpoint
    var queryPath = `/pdf/split?pages=${pages}&url=${uploadedFileUrl}`;
    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);
            if (data.error == false) {
                // Download generated PDF files
                var part = 1;
                data.urls.forEach((url) => {
                    var localFileName = `./part${part}.pdf`;
                    var file = fs.createWriteStream(localFileName);
                    https.get(url, (response2) => {
                        response2.pipe(file)
                        .on("close", () => {
                            console.log(`Generated PDF file saved as "${localFileName}`);
                        });
                    });
                });
            }
        });
    });
}

```

```
        });  
        part++;  
    }, this);  
}  
else {  
    // Service reported error  
    console.log("readBarcodes(): " + data.message);  
}  
});  
})  
.on("error", (e) => {  
    // Request error  
    console.log("readBarcodes(): " + e);  
});  
}
```

---

## 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](http://www.ByteScout.com)

[visit www.PDF.co](http://www.PDF.co)

