

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

Learn in simple ways: How to convert PDF to HTML from uploaded file (node for PDF to HTML API in JavaScript

We regularly create and update our sample code library so you may quickly learn PDF to HTML API and the step-by-step process in JavaScript. ByteScout Cloud API Server helps with PDF to HTML 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!.

JavaScript code snippet like this for ByteScout Cloud API Server works best when you need to quickly implement PDF to HTML API in your JavaScript application. This sample code in JavaScript is all you need. Just copy-paste it to the code editor, then add a reference to ByteScout Cloud API Server and you are ready to try it! Check JavaScript sample code examples to see if they respond to your needs and requirements for the project.

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

Source Code Files:

ConvertPdfToHtmlFromUploadedFile.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
const SourceFile = "./sample.pdf";
// Comma-separated list of page indices (or ranges) to process. Leave empty for all pag
const Pages = "";
// PDF document password. Leave empty for unprotected documents.
const Password = "";
// Destination HTML file name
const DestinationFile = "./result.html";
// Set to `true` to get simplified HTML without CSS. Default is the rich HTML keeping
const PlainHtml = false;
// Set to `true` if your document has the column layout like a newspaper.
const ColumnLayout = false;

// 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 HTML
    convertPdfToHtml(uploadedFileUrl, Password, Pages, PlainHtml, ColumnLayout, Des
  })
  .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: encodeURIComponent(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 convertPdfToHtml(uploadedFileUrl, password, pages, plainHtml, columnLayout, de
// Prepare request to `PDF To HTML` API endpoint
var queryPath = `/pdf/convert/to/html?name=${path.basename(destinationFile)}&password=${password}&simple=${plainHtml}&columns=${columnLayout}&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 HTML file

```

```
    var file = fs.createWriteStream(destinationFile);
    https.get(data.url, (response2) => {
        response2.pipe(file)
        .on("close", () => {
            console.log(`Generated HTML file saved as "${destinationFile}"`);
        });
    });
}
else {
    // Service reported error
    console.log("convertPdfToHtml(): " + data.message);
}
});
})
.on("error", (e) => {
    // Request error
    console.log("convertPdfToHtml(): " + 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)

