

How to add text and images to PDF in JavaScript with ByteScout Cloud API Server

How to write a robust code in JavaScript to add text and images to PDF with this step-by-step tutorial

These sample source codes on this page below are displaying how to add text and images to PDF in JavaScript. Want to add text and images to PDF in your JavaScript app? ByteScout Cloud API Server is designed for it. 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!.

The following code snippet for ByteScout Cloud API Server works best when you need to quickly add text and images to PDF in your JavaScript application. Just copy and paste the code into your JavaScript application's code and follow the instructions. Check JavaScript sample code samples to see if they respond to your needs and requirements for the project.

All these programming tutorials along with source code samples and ByteScout free trial version are available for download from our website.

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:

AddImageToExistingPDF.js

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

// Direct URL of source PDF file.
const SourceFileUrl = "https://bytescout-com.s3.amazonaws.com/files/demo-files/cloud-api/pdf-editor/sample/secure.pdf";

// Comma-separated list of page indices (or ranges) to process. Leave empty for all pages.
const Pages = "";

// PDF document password. Leave empty for unprotected documents.
const Password = "";

// Destination PDF file name
const DestinationFile = "./result.pdf";

// Image params
const Type = "image";
const X = 400;
const Y = 20;
const Width = 119;
const Height = 32;
const ImageUrl = "https://bytescout-com.s3.amazonaws.com/files/demo-files/cloud-api/pdf-editor/sample/secure.pdf";

// Please NOTE: In this sample we're assuming Cloud Api Server is hosted at "https://localhost:8080/"
// If it's not then please replace this with with your hosting url.

// * Add image *
// Prepare request to `PDF Edit` API endpoint
var queryPath = `/pdf/edit/add?name=${path.basename(DestinationFile)}&password=${Password}`;
var reqOptions = {
    host: "localhost",
    path: encodeURI(queryPath)
};

// Send request
https.get(reqOptions, (response) => {
    response.on("data", (d) => {
        // Parse JSON response
        var data = JSON.parse(d);

        if (data.error == false) {
            // Download the PDF file
            var file = fs.createWriteStream(DestinationFile);
            https.get(data.url, (response2) => {
                response2.pipe(file).on("close", () => {
                    console.log(`Generated PDF file saved to '${DestinationFile}' file`);
                });
            });
        }
        else {
            // Service reported error
            console.log(data.message);
        }
    });
});
```

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

www.bytescout.com