

# How to optimize PDF from URL (node for PDF optimization API in JavaScript and PDF.co Web API)

## How to optimize PDF from URL (node for PDF optimization API in JavaScript: How To Tutorial)

Today you are going to learn how to optimize PDF from URL (node in JavaScript). PDF.co Web API helps with PDF optimization API in JavaScript. 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.

The SDK samples like this one below explain how to quickly make your application do PDF optimization API in JavaScript with the help of PDF.co Web API. Follow the instruction and copy - paste code for JavaScript into your project's code editor. Writing JavaScript application typically includes multiple stages of the software development so even if the functionality works please test it with your data and the production environment.

PDF.co Web API - free trial version is on available our website. Also, there are other code samples to help you with your JavaScript application included into trial version.

JavaScript - OptimizePdfFromUrl.js

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

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

// Direct URL of source PDF file.
const SourceFileUrl = "https://bytescout-com.s3.amazonaws.com/files/demo-files/cloud-api/pdf-optimize/sample.pdf";
// PDF document password. Leave empty for unprotected documents.
const Password = "";
// Destination PDF file name
const DestinationFile = "./result.pdf";

// Prepare request to `Optimize PDF` API endpoint
var queryPath = `/v1/pdf/optimize?
name=${path.basename(DestinationFile)}&password=${Password}&url=${SourceFileUrl}&async-
```

```

var reqOptions = {
  host: "api.pdf.co",
  path: encodeURI(queryPath),
  headers: {
    "x-api-key": API_KEY
  }
};
// Send request
https.get(reqOptions, (response) => {
  response.on("data", (d) => {
    // Parse JSON response
    var 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(data.message);
    }
  });
}).on("error", (e) => {
  // Request error
  console.log(e);
});

function checkIfJobIsCompleted(jobId, resultFileUrl) {
  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,
resultFileUrl);}, 3000);
      }
      else if (data.status == "success") {
        // Download PDF file
        var file = fs.createWriteStream(DestinationFile);
        https.get(resultFileUrl, (response2) => {
          response2.pipe(file)
            .on("close", () => {
              console.log(`Generated PDF file saved as
"${DestinationFile}" file.`);
            });
        });
      }
    });
  });
}

```

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

or

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