

[Pro](#) [Teams](#) [Pricing](#) [Documentation](#)**npm**[Sign Up](#)[Sign In](#)[Search](#)

# pdf-parse [DT](#)

1.1.1 • [Public](#) • Published 6 years ago

[Readme](#)[Code](#) [Beta](#)[2 Dependencies](#)[209 Dependents](#)[11 Versions](#)

## pdf-parse

# Pure javascript cross-platform module to extract texts from PDFs.

npm v1.1.1 downloads 19M node >=6.8.1 pipeline passed

# Similar Packages

---

- **pdf2json** buggy, no support anymore, memory leak, throws non-catchable fatal errors
- **j-pdfjson** fork of pdf2json
- **pdf-parser** buggy, no tests
- **pdfreader** using pdf2json
- **pdf-extract** not cross-platform using xpdf

## Installation

---

```
npm install pdf-parse
```

## Basic Usage - Local Files

---

```
const fs = require('fs');
const pdf = require('pdf-parse');

let dataBuffer = fs.readFileSync('path to PDF file...');

pdf(dataBuffer).then(function(data) {

    // number of pages
    console.log(data.numpages);
    // number of rendered pages
    console.log(data.numrender);
    // PDF info
    console.log(data.info);
    // PDF metadata
    console.log(data.metadata);
    // PDF.js version
    // check https://mozilla.github.io/pdf.js/getting\_started/
    console.log(data.version);
    // PDF text
```

```
console.log(data.text);

});
```

## Basic Usage - HTTP

---

You can use **crawler-request** which uses the pdf-parse

## Exception Handling

---

```
const fs = require('fs');
const pdf = require('pdf-parse');

let dataBuffer = fs.readFileSync('path to PDF file...');

pdf(dataBuffer).then(function(data) {
    // use data
})
.catch(function(error){
    // handle exceptions
})
```

## Extend

---

- v1.0.9 and above break pagerender callback **changelog**
- If you need another format like json, you can change page render behaviour with a callback
- Check out <https://mozilla.github.io/pdf.js/>

```
// default render callback
function render_page(pageData) {
    //check documents https://mozilla.github.io/pdf.js/
    let render_options = {
        //replaces all occurrences of whitespace with standard spaces
    }
```

```
        normalizeWhitespace: false,
        //do not attempt to combine same line TextItem's. The default
        disableCombineTextItems: false
    }

    return pageData.getTextContent(render_options)
    .then(function(textContent) {
        let lastY, text = '';
        for (let item of textContent.items) {
            if (lastY == item.transform[5] || !lastY){
                text += item.str;
            }
            else{
                text += '\n' + item.str;
            }
            lastY = item.transform[5];
        }
        return text;
    });
}

let options = {
    pagerender: render_page
}

let dataBuffer = fs.readFileSync('path to PDF file...');

pdf(dataBuffer,options).then(function(data) {
    //use new format
});
```

## Options

---

```
const DEFAULT_OPTIONS = {  
  // internal page parser callback  
  // you can set this option, if you need another format except raw  
  pagerender: render_page,  
  // max page number to parse  
  max: 0,  
  //check https://mozilla.github.io/pdf.js/getting\_started/  
  version: 'v1.10.100'  
}
```

### ***pagerender***(callback)

If you need another format except raw text.

### ***max***(number)

Max number of page to parse. If the value is less than or equal to 0, parser renders all pages.

### ***version***(string, pdf.js version)

check **pdf.js**

- 'default'
- 'v1.9.426'
- 'v1.10.100'
- 'v1.10.88'
- 'v2.0.550'

*default* uses version *v1.10.100*

**[mozilla.github.io/pdf.js](https://mozilla.github.io/pdf.js)**

## Test

- mocha or npm test
- Check **test folder** and **quickstart.js** for extra usages.

# Support

---

I use this package actively myself, so it has my top priority. You can chat on WhatsApp about any infos, ideas and suggestions.

whatsapp chat

## Submitting an Issue

If you find a bug or a mistake, you can help by submitting an issue to [GitLab Repository](#)

## Creating a Merge Request

GitLab calls it merge request instead of pull request.

- [A Guide for First-Timers](#)
- [How to create a merge request](#)
- Check [Contributing Guide](#)

# License

---

**MIT licensed** and all it's dependencies are MIT or BSD licensed.

## Keywords

---

pdf-parse pdf-crawler xpdf pdf.js pdfreader pdf-extractor pdf2json  
j-pdfjson pdf-parser pdf-extract pdf-extractor pdf-to-text pdf-text-extract  
pdfjs server side PDF parsing pdf metadata

## Install

```
> npm i pdf-parse
```



Repository

 [gitlab.com/autokent/pdf-parse](https://gitlab.com/autokent/pdf-parse)

Homepage

 [gitlab.com/autokent/pdf-parse](https://gitlab.com/autokent/pdf-parse)

Weekly Downloads

415,010



Version

1.1.1

License

MIT

Unpacked Size

33.3 MB

Total Files

770

Last publish

6 years ago

Collaborators



[Try on RunKit](#)

[Report malware](#)





## Support

[Help](#)

[Advisories](#)

[Status](#)

[Contact npm](#)

## Company

[About](#)

[Blog](#)

[Press](#)

## Terms & Policies

[Policies](#)

[Terms of Use](#)

[Code of Conduct](#)

[Privacy](#)