

MDPDF - Markdown to PDF converter

build passing build passing npm v2.0.1 code style prettier

A command line markdown to pdf converter with support for page headers, footers, and custom stylesheets. Mdpdf is incredibly configurable and has a JavaScript API for more extravagant usage.

For examples of how to use headers and footers, see the [examples directory](#).

If you're using the [Atom text editor](#), checkout the [markdown-pdf](#) plugin which uses mdpdf.

Donate

This project is actively maintained during evenings and weekends, please consider either donating or contributing in other ways.



If you wish to report bugs or contribute fixes and features, please see the [contributors guide](#)

Installation

Install globally to use from the command line.

```
npm install mdpdf -g
```

Install locally to access the API.

```
npm install mdpdf --save
```

Example usage

- `mdpdf README.md` - Simple convert using GitHub Markdown CSS and some additional basic stylings.
- `mdpdf README.md --style styles.css --header header.hbs --height 22` - Convert with custom styling with a header of height 22mm.

Options

- `--style=<filename>` - A single css stylesheet you wish to apply to the PDF
- `--header=<filename>` - A HTML (.html) file to inject into the header of the PDF
- `--h-height=<height>` - The height of the header section
- `--footer=<filename>` - A HTML (.html) file to inject into the footer of the PDF
- `--f-height=<height>` - The height of the footer section
- `--border=<size>` - Border (top, left, bottom, right; default: 20mm)
- `--border-top=<size>` - Top border (default: 20mm)
- `--border-left=<size>` - Left border (default: 20mm)
- `--border-bottom=<size>` - Bottom border (default: 20mm)
- `--border-right=<size>` - Right border (default: 20mm)
- `--no-emoji` - Disables emoji conversions
- `--debug` - Save the generated html for debugging

- `--help` - Display this menu
- `--version` - Display the application version
- `--format=<format>` - PDF size format: A3, A4, A5, Legal, Letter, Tabloid (Default: A4)
- `--orientation=<orientation>` - PDF orientation: portrait or landscape (Default: portrait)

Length parameters (`<height>` and `<size>`) require a unit. Valid units are `mm` , `cm` , `in` and `px` .

Headers and footers

Mdpdf is powered by Puppeteer, the headless Chrome browser project by Google. [Puppeteer provides a number of header and footer html classes which can be used to insert things such as page numbers.](#)

Note: mdpdf pre-2.x made use of Phantom.js which had considerably better support for headers and footers, and including better styling support for them. Sadly Phantom.js is no longer supported and had a number of other rendering bugs meaning it was no longer possible to support it as a core component of mdpdf. If you previously relied on good header and footer support you may wish to stick with 1.x releases until Puppeteer prioritises better header and footer support. Currently headers and footers do work in 2.x+ releases, but their styles must be handled independently of the main markdown file via `--styles` options and a few css tags do not work correctly. Past this there shouldn't be any issues with 2.x+ headers and footers.

Environment variables

- `MDPDF_STYLES` - The full path to a stylesheet you wish to use if `--style` is not specified when calling `mdpdf` from the command line.

Emoji support

In text emoji's are also supported, but there are a few instances of shorthand which do not work and require the longhand version, i.e. `:+1:` doesn't work but `👍` will.

Programatic API

The API is very straight forward with a single function `convert()` which takes some options. The convert method uses a promise generated by the [Bluebird.js](#) library. For a full example see the [bin/index.js!](#)

Example API usage

```
const mdpdf = require('mdpdf');
const path = require('path');

let options = {
  source: path.join(__dirname, 'README.md'),
  destination: path.join(__dirname, 'output.pdf'),
  styles: path.join(__dirname, 'md-styles.css'),
  pdf: {
    format: 'A4',
    orientation: 'portrait'
  }
};

mdpdf.convert(options).then((pdfPath) => {
  console.log('PDF Path:', pdfPath);
}).catch((err) => {
  console.error(err);
});
```

Options

- `source` (**required**) - Full path to the source markdown file.
- `destination` (**required**) - Full path to the destination (pdf) file.
- `styles` - Full path to a single css stylesheet which is applied last to the PDF.
- `ghStyle` - Boolean value of whether or not to use the GitHub Markdown CSS, *defaults to false*.
- `defaultStyle` - Boolean value of whether or not to use the additional default styles. These styles provide some things like a basic border and font size. *Defaults to false*.
- `header` - Full path to a the Handlebars (`.hbs`) file which will be your header. If you set this, you must set the header height (see below).
- `debug` - When this is set the intermediate HTML will be saved into a file.
- `pdf` (**required**) - An object which contains some sub parameters to control the final PDF document
 - `format` (**required**) - Final document format, allowed values are "A3, A4, A5, Legal, Letter, Tabloid"
 - `orientation` - Final document size orientation, allowed values are "potrait, orientation"
 - `header` - A sub object which contains some header settings
 - `height` - Height of the documents header in mm (default 45mm). If you wish to use a header, then this must be set.
 - `border` - The document borders