

ProductsPricingDocumentationCommunity

npm

Sign UpSign In

Search packages

Search

Get unlimited public & private packages + team-based management with npm Teams. [Learn more »](#)

# html-pdf

2.2.0 • Public • Published 2 years ago

 [Readme](#)

 [Explore](#) BETA

 [1 Dependency](#)

 [227 Dependents](#)

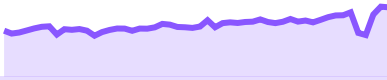
 [21 Versions](#)

## Install

```
> npm i html-pdf
```

Weekly Downloads

81,718



Version

2.2.0

License

MIT

Issues

349

Pull Requests

12

### Homepage

 [github.com/marcbachmann/node-html-pdf](https://github.com/marcbachmann/node-html-pdf)

---

### Repository

 [github.com/marcbachmann/node-html-pdf](https://github.com/marcbachmann/node-html-pdf)

---

### Last publish

2 years ago

---

### Collaborators



[> Try on RunKit](#)

[🚩 Report a vulnerability](#)

# node-html-pdf

---

## HTML to PDF converter that uses phantomjs

---



**Example Business Card**

-> **and its Source file**

**Example Receipt**

## Changelog

---

Have a look at the releases page: <https://github.com/marcbachmann/node-html-pdf/releases>

## Installation

---

Install the html-pdf utility via **npm**:

```
$ npm install -g html-pdf
```

## Command-line example

---

```
$ html-pdf test/businesscard.html businesscard.pdf
```

## Code example

---

```
var fs = require('fs');
var pdf = require('html-pdf');
var html = fs.readFileSync('./test/businesscard.html', 'utf8');
var options = { format: 'Letter' };

pdf.create(html, options).toFile('./businesscard.pdf', function(err, res){
  if (err) return console.log(err);
  console.log(res); // { filename: '/app/businesscard.pdf' }
});
```



## API

---

```
var pdf = require('html-pdf');
pdf.create(html).toFile([filepath, ],function(err, res){
  console.log(res.filename);
});

pdf.create(html).toStream(function(err, stream){
  stream.pipe(fs.createWriteStream('./foo.pdf'));
});

pdf.create(html).toBuffer(function(err, buffer){
  console.log('This is a buffer:', Buffer.isBuffer(buffer));
});
```

```
// for backwards compatibility
// alias to pdf.create(html[, options]).toBuffer(callback)
pdf.create(html [, options], function(err, buffer){});
```

## Footers and Headers

html-pdf can read the header or footer either out of the footer and header config object or out of the html source. You can either set a default header & footer or overwrite that by appending a page number (1 based index) to the id="pageHeader" attribute of a html tag.

You can use any combination of those tags. The library tries to find any element, that contains the pageHeader or pageFooter id prefix.

```
<div id="pageHeader">Default header</div>
<div id="pageHeader-first">Header on first page</div>
<div id="pageHeader-2">Header on second page</div>
<div id="pageHeader-3">Header on third page</div>
<div id="pageHeader-last">Header on last page</div>
...
<div id="pageFooter">Default footer</div>
<div id="pageFooter-first">Footer on first page</div>
<div id="pageFooter-2">Footer on second page</div>
<div id="pageFooter-last">Footer on last page</div>
```

## Options

---

```
config = {

  // Export options
  "directory": "/tmp",           // The directory the file gets written to

  // Papersize Options: http://phantomjs.org/api/webpage/properties/
  "height": "10.5in",           // allowed units: mm, cm, in, px
  "width": "8in",               // allowed units: mm, cm, in, px
  - or -
```

```

"format": "Letter",          // allowed units: A3, A4, A5, Legal
"orientation": "portrait", // portrait or landscape

// Page options
"border": "0",              // default is 0, units: mm, cm, in
- or -
"border": {
  "top": "2in",             // default is 0, units: mm, cm, in
  "right": "1in",
  "bottom": "2in",
  "left": "1.5in"
},

paginationOffset: 1,        // Override the initial pagination
"header": {
  "height": "45mm",
  "contents": '<div style="text-align: center;">Author: Marc
},
"footer": {
  "height": "28mm",
  "contents": {
    first: 'Cover page',
    2: 'Second page', // Any page number is working. 1-based
    default: '<span style="color: #444;">{{page}}</span><span>
    last: 'Last Page'
  }
},

// Rendering options
"base": "file:///home/www/your-asset-path", // Base path that
// Zooming option, can be used to scale images if `options.zoomFactor`
"zoomFactor": "1", // default is 1

// File options
"type": "pdf",          // allowed file types: png, jpeg,

```

```
"quality": "75", // only used for types png & jpeg

// Script options
"phantomPath": "./node_modules/phantomjs/bin/phantomjs", //
"phantomArgs": [], // array of strings used as phantomjs arguments
"script": '/url', // Absolute path to a custom phantomjs script
"timeout": 30000, // Timeout that will cancel phantomjs if it takes longer than this time

// Time we should wait after window load
// accepted values are 'manual', some delay in milliseconds
"renderDelay": 1000,

// HTTP Headers that are used for requests
"httpHeaders": {
  // e.g.
  "Authorization": "Bearer ACEFAD8C-4B4D-4042-AB30-6C735F5BA",
},

// To run Node application as Windows service
"childProcessOptions": {
  "detached": true
}

// HTTP Cookies that are used for requests
"httpCookies": [
  // e.g.
  {
    "name": "Valid-Cookie-Name", // required
    "value": "Valid-Cookie-Value", // required
    "domain": "localhost",
    "path": "/foo", // required
    "httponly": true,
    "secure": false,
    "expires": (new Date()).getTime() + (1000 * 60 * 60) //
  }
]
```

```
}
```

The full options object gets converted to JSON and will get passed to the phantomjs script as third argument.

There are more options concerning the paperSize, header & footer options inside the phantomjs script.

## Keywords

---

**html pdf phantom phantomjs nodejs**



## Help

[Documentation](#)

[Community](#)

[Resources](#)

[Advisories](#)

[Status](#)

[Contact](#)

## About

[Company](#)

[Blog](#)

[Careers](#)

[Webinars](#)

[Press](#)

[Newsletter](#)

## **Terms & Policies**

[Policies](#)

[Terms of Use](#)

[Code of Conduct](#)