

marqius

REPLACE WKHTMLTOPDF

GETTING STARTED

Keep your commands. Change the engine.

LINUX · MAC · DOCKER · TEN MINUTES

MARQIUS-WKHTMLTOPDF 0.12.2 · NOVEMBER 2026

Getting started with marqius-wkhtmltopdf

marqius-wkhtmltopdf takes the place of wkhtmltopdf. Your app, your HTML templates and your wrapper library (pdfkit, Snappy, laravel-snappy, wicked_pdf, Rotativa and the rest) stay as they are: the same commands, the same options, the same exit codes.

This guide takes about ten minutes. It starts with the commands you already use, then installs the program, checks that everything works, and switches your app over. Every command in it was run as printed, and all of them are attached to this PDF as `commands.txt`.

CONTENTS

1 The commands you already use 3
2 Install it 4
3 Check that everything works 6
4 Switch your app 8
5 Keep it up to date, or go back 9
6 If something is not right 10
7 Licence and next steps 10

1 The commands you already use

Every wkhtmltopdf command works unchanged. These are the ones teams use most. `invoice.html` and `report.html` stand for your own pages.

Make a PDF

From a file, from a web address, and from standard input to standard output, as many wrapper libraries call it:

TERMINAL

```
wkhtmltopdf invoice.html invoice.pdf
wkhtmltopdf https://example.com example.pdf
wkhtmltopdf - - < invoice.html > invoice.pdf
```

Paper, orientation and margins

A4 is the default, as in wkhtmltopdf.

TERMINAL

```
wkhtmltopdf -s Letter -O Landscape report.html report.pdf
wkhtmltopdf -T 20mm -B 20mm -L 15mm -R 15mm invoice.html invoice.pdf
```

Page numbers, headers and footers

TERMINAL

```
wkhtmltopdf --footer-center "Page [page] of [topage]" report.html report.pdf
wkhtmltopdf --header-left "Sales report" --header-right "[date]" report.html report.pdf
```

`[page]`, `[topage]`, `[date]`, `[time]` and `[title]` are filled in on every page, as wkhtmltopdf fills them.

A table of contents, and the document's title

TERMINAL

```
wkhtmltopdf toc report.html report.pdf
wkhtmltopdf --title "Monthly report" report.html report.pdf
```

Pages that need a little more

TERMINAL

```
wkhtmltopdf --enable-local-file-access invoice.html invoice.pdf
wkhtmltopdf --javascript-delay 1000 report.html report.pdf
wkhtmltopdf --print-media-type report.html report.pdf
wkhtmltopdf -q invoice.html invoice.pdf
```

- `--enable-local-file-access` lets the page use pictures and styles from local files. Without it they are blocked, with wkhtmltopdf 0.12.6's own message, as before.
- `--javascript-delay 1000` waits a second for the page's scripts.
- `--print-media-type` uses the page's print styles.
- `-q` leaves out the progress lines.

Pictures

TERMINAL

```
wkhtmltoimage report.html report.png
wkhtmltoimage --width 1200 --format jpg report.html report.jpg
```

2
Install it

On a Linux server or a Mac

On a Linux server (Intel/AMD or ARM) or a Mac with Apple silicon, one line installs it:

TERMINAL

```
curl -fsSL https://marqius.com/marqius-wkhtmltopdf.sh | bash
```

Terminal — a Linux server

```
$ curl -fsSL https://marqius.com/marqius-wkhtmltopdf.sh | bash

  marqius-wkhtmltopdf - the maintained drop-in for wkhtmltopdf

  + Platform     linux arm64
  + Version      0.12.2 (latest)
  + Downloaded   marqius-wkhtmltopdf-linux-arm64, 16 MB, checksum verified
  + Installed    /usr/local/bin/marqius-wkhtmltopdf
  + wkhtmltopdf  /usr/local/bin/wkhtmltopdf and wkhtmltoimage run marqius-wkhtmltopdf

  marqius-wkhtmltopdf 0.12.2 is installed. Programs that run wkhtmltopdf now run it.

  Next steps
    1. Check the version your libraries will see
       wkhtmltopdf -V
    2. Make a PDF the way your app does
       wkhtmltopdf --page-size A4 page.html page.pdf
    3. Compare it with a PDF wkhtmltopdf made before
       marqius-wkhtmltopdf compare old.pdf page.pdf

  Update   run the same command again
  Undo     run it again, ending  bash -s -- --uninstall
  Guide    https://marqius.com/download/marqius-wkhtmltopdf/replace-wkhtmltopdf.pdf
  Help     hello@marqius.com
```

It downloads the latest version, checks it against its published checksum, and installs it in `/usr/local/bin` under its own name and the names `wkhtmltopdf` and `wkhtmltoimage`. For a user without administrator rights it goes into `~/.local/bin`, and the installer prints the line that puts that folder on your PATH.

If wkhtmltopdf is installed already, nothing is replaced until you say so. This line puts marqius-wkhtmltopdf in its place and keeps the old program, so you can always go back:

TERMINAL

```
curl -fsSL https://marqius.com/marqius-wkhtmltopdf.sh | bash -s -- --take-over
```

With Docker

TERMINAL

```
docker pull marqius/wkhtmltopdf
```

The image holds the program and its fonts, ready to run. Use any command of chapter 1 with `docker run --rm -v "$PWD:/data" marqius/wkhtmltopdf` in place of `wkhtmltopdf`:

TERMINAL

```
docker run --rm -v "$PWD:/data" marqius/wkhtmltopdf invoice.html invoice.pdf
```

TAG	WHAT IT IS
`latest`	The program and its fonts: for most teams
`slim`	The smallest image, with one font (Liberation Sans)
`alpine`, `trixie`, `bookworm`, `bullseye`	The same on Alpine or Debian 13, 12 or 11, to build your own image on

3 Check that everything works

Five checks, about a minute in all.

1. **Your apps will run marqius-wkhtmltopdf.**

   TERMINAL

   ```
   wkhtmltopdf -V
   ```

   It answers `wkhtmltopdf 0.12.6 (compatible; marqius-wkhtmltopdf 0.12.2)`. Wrapper libraries read the first part; the part in brackets shows that marqius-wkhtmltopdf answered. With Docker, `docker run --rm marqius/wkhtmltopdf -V` answers `marqius-wkhtmltopdf 0.12.2`.

2. **It is where your apps look.**

   TERMINAL

   ```
   command -v wkhtmltopdf
   ```

   It answers `/usr/local/bin/wkhtmltopdf`, or the same in `~/.local/bin`. If your app names another path, chapter 4 shows where to change it.

3. **It makes a PDF.** No file of your own is needed:

   TERMINAL

   ```
   echo '<h1>It works</h1>' | wkhtmltopdf - it-works.pdf
   ```

   The last line reads `Done`. Open `it-works.pdf`: it says "It works".

Terminal — a Linux server

```
$ wkhtmltopdf -V
wkhtmltopdf 0.12.6 (compatible; marqius-wkhtmltopdf 0.12.2)
$ echo '<h1>It works</h1>' | wkhtmltopdf - it-works.pdf
Loading pages (1/6)
Counting pages (2/6)
Resolving links (4/6)
Loading headers and footers (5/6)
Printing pages (6/6)
Done
$ wkhtmltopdf https://example.com example.pdf
Loading pages (1/6)
Counting pages (2/6)
Resolving links (4/6)
Loading headers and footers (5/6)
Printing pages (6/6)
Done
```

On a new server that has no fonts yet, it stops instead and prints the line that installs them on that very system. Run that line, then the check again. On Debian or Ubuntu it is:

TERMINAL

```
sudo apt-get update
sudo apt-get install -y --no-install-recommends fonts-liberation2 fonts-dejavu-core
```

4. **Your documents come out as before.** Make one of your PDFs the new way, and compare it with the same PDF from wkhtmltopdf:

TERMINAL

```
marqius-wkhtmltopdf compare old.pdf new.pdf
```

Terminal — a Linux server

```
$ marqius-wkhtmltopdf compare old.pdf new.pdf
old.pdf  against  new.pdf
verdict: appearance only (review)
pages 1 and 1, words 99 and 99

how it looks: 1 difference
  moved           page 1   the line "Due: 17 Dec 2026" moved 1 pt to the left

the file itself: 1 difference
  pdf-type                 the file is PDF 1.7, was PDF 1.4  [version]

note: the old file was made by wkhtmltopdf 0.12.6 / Qt 4.8.7; the new one by marqius /
marqius 0.12.0 (not counted as a difference)
```

The verdict is `same`, `appearance only` (the same words, with something moved or looking a little different) or `content differs` (words, numbers or pictures changed). To check many documents at once, put the old PDFs in one folder and the new ones, with the same names, in another:

TERMINAL

```
marqius-wkhtmltopdf compare old/ new/ -o report/
```

`report/` then holds a summary of every pair, and copies of both PDFs with each difference marked.

5. **Your app makes its PDFs.** Make a PDF in your app the way your users do, and open it. In the PDF's properties, in any PDF viewer, the producer reads `marqius 0.12.2`.

4 Switch your app

Most apps run `wkhtmltopdf` by its name and find it on the PATH. After `--take-over`, or on a server that had no wkhtmltopdf, restart your app: it now runs marqius-wkhtmltopdf.

If your app names a path, point it at `/usr/local/bin/wkhtmltopdf`:

LIBRARY	WHERE THE PATH GOES
pdfkit (Python)	`pdfkit.configuration(wkhtmltopdf="/usr/local/bin/wkhtmltopdf")`
laravel-snappy (PHP)	`WKHTML_PDF_BINARY=/usr/local/bin/wkhtmltopdf` in `.env`
wicked_pdf (Ruby)	`WickedPdf.config = { exe_path: "/usr/local/bin/wkhtmltopdf" }`
wkhtmltopdf (Node.js)	`wkhtmltopdf.command = "/usr/local/bin/wkhtmltopdf";`
go-wkhtmltopdf (Go)	`wkhtmltopdf.SetPath("/usr/local/bin/wkhtmltopdf")`

Rotativa, Wkhtmltopdf.NetCore and the other libraries are in the switching manual.

In your own Docker image

Two lines in your Dockerfile copy in the program and its fonts:

DOCKERFILE

```
COPY --from=marqius/wkhtmltopdf /bin/wkhtmltopdf /usr/local/bin/wkhtmltopdf
COPY --from=marqius/wkhtmltopdf /usr/share/fonts /usr/share/fonts
```

Then delete the lines that installed wkhtmltopdf and its libraries (libx11, libxrender, libxext, libssl, fontconfig, freetype): marqius-wkhtmltopdf is one program and needs none of them.

As a web service

One line starts a PDF service; your apps send the HTML and get the PDF back:

TERMINAL

```
docker run -d -p 8080:80 marqius/wkhtmltopdf serve
curl localhost:8080/health
echo '<h1>It works</h1>' | curl -F file=@- localhost:8080 -o it-works.pdf
```

`ok` means the service is running. It answers the same requests as the wkhtmltopdf-as-a-service images (openlabs/docker-wkhtmltopdf-aas and those built on it), so if you use one, change only the image's name. Keep it on your private network.

Terminal — Docker

```
$ docker run --rm marqius/wkhtmltopdf -V
marqius-wkhtmltopdf 0.12.2
$ echo '<h1>It works</h1>' | docker run --rm -i marqius/wkhtmltopdf - - > it-works.pdf
Loading pages (1/6)
Counting pages (2/6)
Resolving links (4/6)
Loading headers and footers (5/6)
Printing pages (6/6)
Done
$ docker run --rm -v "$PWD:/data" marqius/wkhtmltopdf invoice.html invoice.pdf
Loading pages (1/6)
Counting pages (2/6)
Resolving links (4/6)
Loading headers and footers (5/6)
Printing pages (6/6)
Done
```

5
Keep it up to date, or go back

On a server, the install line updates too. The version before is kept, so you can return to it, or take everything away:

TERMINAL

```
curl -fsSL https://marqius.com/marqius-wkhtmltopdf.sh | bash
curl -fsSL https://marqius.com/marqius-wkhtmltopdf.sh | bash -s -- --rollback
curl -fsSL https://marqius.com/marqius-wkhtmltopdf.sh | bash -s -- --uninstall
```

After `--uninstall`, the wkhtmltopdf you had before answers again.

With Docker, pull the image, then start your containers again or rebuild your images. Docker keeps using the copy it has until you pull:

TERMINAL

```
docker pull marqius/wkhtmltopdf
```

6 If something is not right

WHAT YOU SEE	WHAT TO DO
`Error: no fonts are installed here`	Run the line it prints: it is the one for your system. Then try again.
`wkhtmltopdf -V` **does not mention marqius**	Your app still finds the old wkhtmltopdf. Install with `--take-over`, or point your app at `/usr/local/bin/wkhtmltopdf`.
`command not found`	It was installed for your user only: run the `export PATH=...` line the installer printed.
`Blocked access to file`	The page uses local pictures or styles: add `--enable-local-file-access`, as with wkhtmltopdf 0.12.6.
Docker on Linux: the PDF belongs to root	Add `--user "$(id -u):$(id -g)"` after `docker run --rm`.
Docker: an older version answers	Run `docker pull marqius/wkhtmltopdf`, then start the container again.
Anything else	Write to hello@marqius.com with the command and what it printed.

7 Licence and next steps

marqius-wkhtmltopdf is free for 30 days. Plans and licences: marqius.com/licence/marqius-wkhtmltopdf.

- **The switching manual**, with every wrapper library, Docker, fonts and security: marqius.com/download/replace-wkhtmltopdf.pdf
- **The Docker guide**: marqius.com/wkhtmltopdf/docker
- **Help**: hello@marqius.com