Package 'cpp11qpdf'

May 8, 2026

Type Package

Title Split, Combine and Compress PDF Files

Version 1.3.5

Description Bindings to 'qpdf':
'qpdf' (<https://qpdf.sourceforge.io/>) is a an open-source PDF rendering
library that allows to conduct content-preserving transformations of PDF
files such as split, combine, and compress PDF files.

License Apache License (>= 2)

URL https://pacha.dev/cpp11qpdf/

BugReports https://github.com/pachadotdev/cpp11qpdf/issues

SystemRequirements libjpeg (deb: libjpeg-dev, rpm: libjpeg-devel,
brew: libjpeg)

Imports curl

LinkingTo cpp11

RoxygenNote 7.3.2

Suggests knitr, rmarkdown, spelling, testthat (>= 3.0.0)

Encoding UTF-8

Language en-US

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation yes

Author Mauricio Vargas Sepulveda [aut, cre] (ORCID:
<https://orcid.org/0000-0003-1017-7574>),
Jeroen Ooms [aut] (Author of qpdf R package, ORCID:
<https://orcid.org/0000-0002-4035-0289>),
Ben Raymond [ctb],
Jay Berkenbilt [cph] (Author of qpdf),
Munk School of Global Affairs and Public Policy [fnd]

Maintainer Mauricio Vargas Sepulveda <m.sepulveda@mail.utoronto.ca>

Repository CRAN

Date/Publication 2024-12-19 20:40:09 UTC

Contents

`cpp11qpdf-package` *Split, Combine and Compress PDF Files*

Description

Bindings to 'cpp11qpdf': A tool for content-preserving transformations of PDF files such as split, combine, and compress.

Author(s)

Maintainer: Mauricio Vargas Sepulveda <m.sepulveda@mail.utoronto.ca> (ORCID)

Authors:

- Jeroen Ooms <jeroen@berkeley.edu> (ORCID) (Author of qpdf R package)

Other contributors:

- Ben Raymond [contributor]
- Jay Berkenbilt (Author of qpdf) [copyright holder]
- Munk School of Global Affairs and Public Policy [funder]

See Also

Useful links:

- https://pacha.dev/cpp11qpdf/
- Report bugs at https://github.com/pachadotdev/cpp11qpdf/issues

pdf_combine *Combine multiple pdf files into a single pdf file*

Description

Combine multiple pdf files into a single pdf file

Usage

```
pdf_combine(input, output = NULL, password = "")
```

Arguments

input	path or url to the input pdf file
output	base path of the output file(s)
password	string with password to open pdf file

Value

a character vector with the path of the combined pdf file

Examples

```
pdf_file <- system.file("examples", "sufganiyot.pdf", package = "cpp11qpdf")
fout <- tempfile()
pdf_combine(pdf_file, fout, "")
```

pdf_compress *Compress a pdf file*

Description

Compress a pdf file

Usage

```
pdf_compress(input, output = NULL, linearize = FALSE, password = "")
```

Arguments

input	path or url to the input pdf file
output	base path of the output file(s)
linearize	enable pdf linearization (streamable pdf)
password	string with password to open pdf file

Value

a character vector with the path of the compressed pdf file

Examples

```
pdf_file <- system.file("examples", "sufganiyot.pdf", package = "cpp11qpdf")
fout <- tempfile()
pdf_compress(pdf_file, fout, TRUE, "")
```

`pdf_length` *Get the number of pages in a pdf file*

Description

Get the number of pages in a pdf file

Usage

```
pdf_length(input, password = "")
```

Arguments

`input`	path or url to the input pdf file
`password`	string with password to open pdf file

Value

an integer value with the number of pages in the pdf file

Examples

```
pdf_file <- system.file("examples", "sufganiyot.pdf", package = "cpp11qpdf")
pdf_length(pdf_file, "")
```

`pdf_overlay_stamp` *Overlay a pdf file into another pdf file*

Description

Overlay a pdf file into another pdf file

Usage

```
pdf_overlay_stamp(input, stamp, output = NULL, password = "")
```

Arguments

input	path or url to the input pdf file
stamp	pdf file of which the first page is overlayed into each page of input
output	base path of the output file(s)
password	string with password to open pdf file

Value

a character vector with the path of the stamped pdf file

Examples

```
pdf_file <- system.file("examples", "sufganiyot.pdf", package = "cpp11qpdf")
stamp_file <- system.file("examples", "header.pdf", package = "cpp11qpdf")
fout <- tempfile()
pdf_overlay_stamp(pdf_file, stamp_file, fout, "")
```

---

pdf_rotate_pages *Rotate pages in a pdf file*

Description

Rotate pages in a pdf file

Usage

```
pdf_rotate_pages(
  input,
  pages,
  angle = 90,
  relative = FALSE,
  output = NULL,
  password = ""
)
```

Arguments

input	path or url to the input pdf file
pages	a vector with page numbers to rotate
angle	rotation angle in degrees (positive = clockwise)
relative	if TRUE, pages are rotated relative to their current orientation. If FALSE, rotation is absolute (0 = portrait, 90 = landscape, rotated 90 degrees clockwise from portrait)
output	base path of the output file(s)
password	string with password to open pdf file

Value

a character vector with the path of the rotated pdf file

Examples

```
pdf_file <- system.file("examples", "sufganiyot.pdf", package = "cpp11qpdf")
fout <- tempfile()
pdf_rotate_pages(pdf_file, 1, 90, FALSE, fout, "")
```

---

pdf_split *Split a pdf file into individual pages*

Description

Split a pdf file into individual pages

Usage

```
pdf_split(input, output = NULL, password = "")
```

Arguments

input	path or url to the input pdf file
output	base path of the output file(s)
password	string with password to open pdf file

Value

a character vector with the paths of the split pdf files

Examples

```
# extract some pages
pdf_file <- system.file("examples", "sufganiyot.pdf", package = "cpp11qpdf")
fout <- tempfile()
pdf_split(pdf_file, fout, password = "")
```

pdf_subset *Subset a pdf file to a new pdf file containing the selected pages*

Description

Subset a pdf file to a new pdf file containing the selected pages

Usage

```
pdf_subset(input, pages = 1, output = NULL, password = "")
```

Arguments

input	path or url to the input pdf file
pages	a vector with page numbers to select. Negative numbers means removing those pages (same as R indexing)
output	base path of the output file(s)
password	string with password to open pdf file

Value

a character vector with the path of the subsetted pdf file

Examples

```
pdf_file <- system.file("examples", "sufganiyot.pdf", package = "cpp11qpdf")
fout <- tempfile()
pdf_subset(pdf_file, 1, fout, "")
```

Index