

# Imageboxes using optimized json

border:5

w=100, h=200

h=200

pdfcpu is a tool for PDF manipulation written in Go.

Usage:

pdfcpu command [arguments]

The commands are:

validate	validate PDF against PDF 32000-1:2008 (PDF 1.7)
optimize	optimize PDF by getting rid of redundant page resources
split	split multi-page PDF into several single-page PDFs
merge	concatenate 2 or more PDFs
extract	extract images, fonts, content, pages out of a PDF
trim	create trimmed version of a PDF
version	print pdfcpu version

Single-letter Unix-style supported for commands and flags.

Use "pdfcpu help [command]" for more information about a command.

w=200, h=100

pdfcpu is a tool for PDF manipulation written in Go.

Usage:

pdfcpu command [arguments]

The commands are:

validate	validate PDF against PDF 32000-1:2008 (PDF 1.7)
optimize	optimize PDF by getting rid of redundant page resources
split	split multi-page PDF into several single-page PDFs
merge	concatenate 2 or more PDFs
extract	extract images, fonts, content, pages out of a PDF
trim	create trimmed version of a PDF
version	print pdfcpu version

Single-letter Unix-style supported for commands and flags.

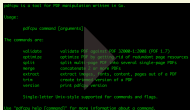
Use "pdfcpu help [command]" for more information about a command.

w=200

# Imageboxes using optimized json

bgCol, border:5, padding: 10

w=100, h=200



h=200

```
pdfcpu is a tool for PDF manipulation written in Go.

Usage:

    pdfcpu command [arguments]

The commands are:

    validate    validate PDF against PDF 32000-1:2008 (PDF 1.7)
    optimize    optimize PDF by getting rid of redundant page resources
    split       split multi-page PDF into several single-page PDFs
    merge       concatenate 2 or more PDFs
    extract     extract images, fonts, content, pages out of a PDF
    trim        create trimmed version of a PDF
    version     print pdfcpu version

Single-letter Unix-style supported for commands and flags.

Use "pdfcpu help [command]" for more information about a command.
```

w=200, h=100

```
pdfcpu is a tool for PDF manipulation written in Go.

Usage:

    pdfcpu command [arguments]

The commands are:

    validate    validate PDF against PDF 32000-1:2008 (PDF 1.7)
    optimize    optimize PDF by getting rid of redundant page resources
    split       split multi-page PDF into several single-page PDFs
    merge       concatenate 2 or more PDFs
    extract     extract images, fonts, content, pages out of a PDF
    trim        create trimmed version of a PDF
    version     print pdfcpu version

Single-letter Unix-style supported for commands and flags.

Use "pdfcpu help [command]" for more information about a command.
```

```
pdfcpu is a tool for PDF manipulation written in Go.

Usage:

    pdfcpu command [arguments]

The commands are:

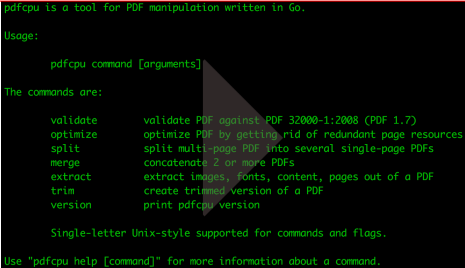
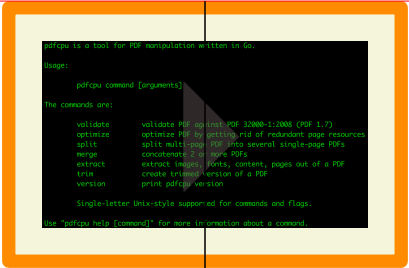
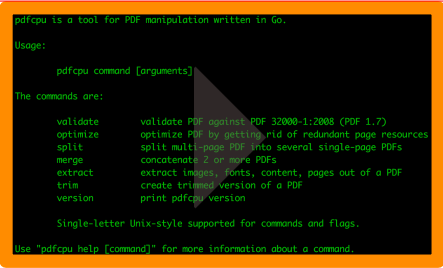
    validate    validate PDF against PDF 32000-1:2008 (PDF 1.7)
    optimize    optimize PDF by getting rid of redundant page resources
    split       split multi-page PDF into several single-page PDFs
    merge       concatenate 2 or more PDFs
    extract     extract images, fonts, content, pages out of a PDF
    trim        create trimmed version of a PDF
    version     print pdfcpu version

Single-letter Unix-style supported for commands and flags.

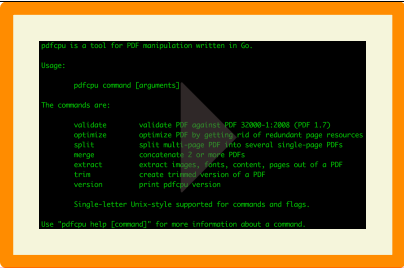
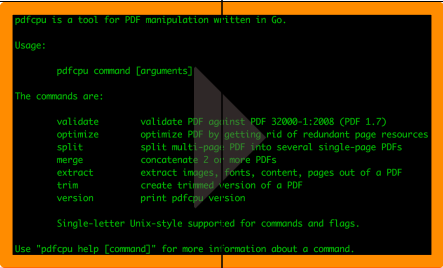
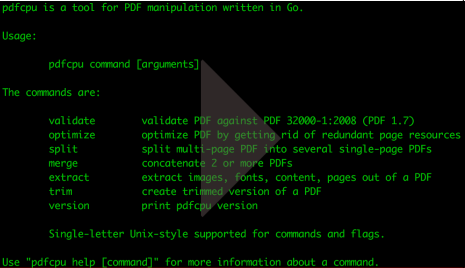
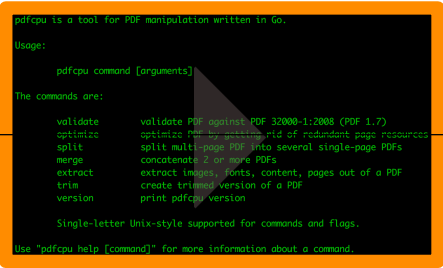
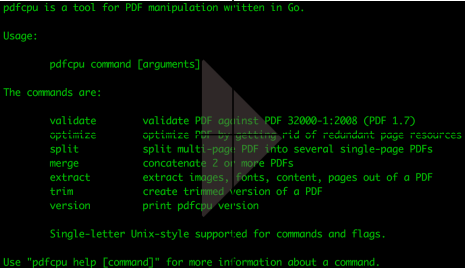
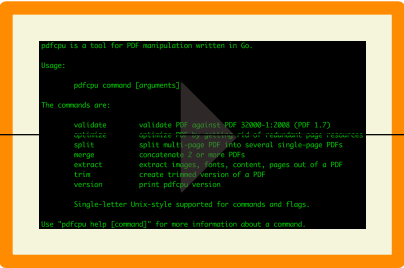
Use "pdfcpu help [command]" for more information about a command.
```

w=200

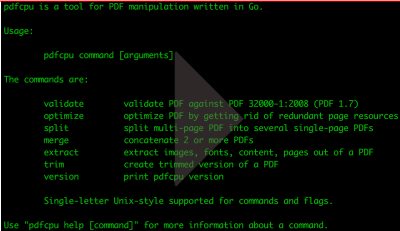
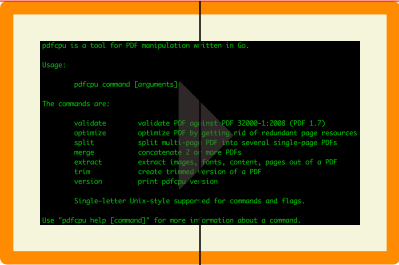
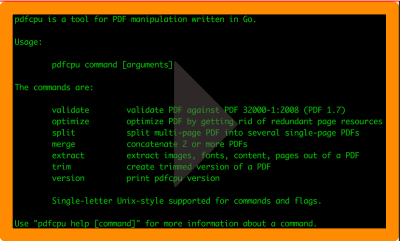
# Imageboxes using optimized json



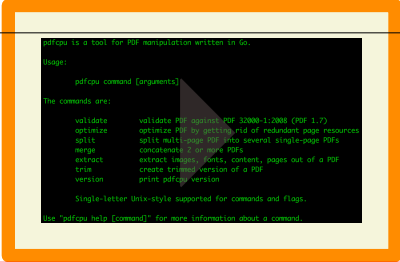
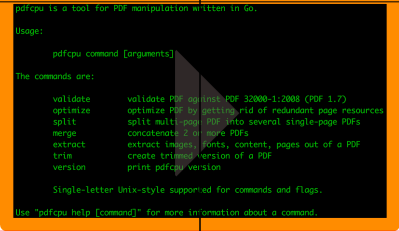
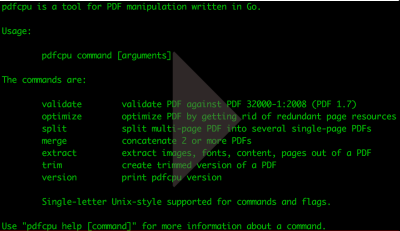
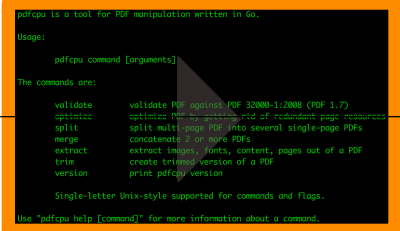
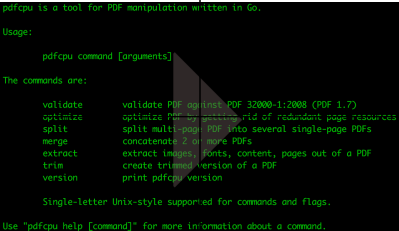
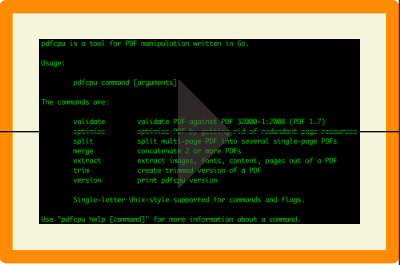
anchored, height: 100



# Imageboxes using optimized json



anchored, width: 150



# Imageboxes using optimized json

anchored, width: 100, height: 100

