jfscripts

Release 0.0.0

Jul 11, 2022

Contents

1 Comande line interfaces **3**
- 1.1 dns-ipv6-prefix.py . . . 3
- 1.2 extract-pdftext.py . . . 3
- 1.3 find-dupes-by-size.py . . . 4
- 1.4 list-files.py . . . 4
- 1.5 mac-to-eui64.py . . . 5
- 1.6 pdf-compress.py . . . 5
- 1.7 image-into-pdf.py . . . 9

2 jfscripts **11**
- 2.1 jfscripts package . . . 11

3 Indices and tables **19**

4 jfscripts **21**
- 4.1 dns-ipv6-prefix.py . . . 21
- 4.2 extract-pdftext.py . . . 21
- 4.3 find-dupes-by-size.py . . . 22
- 4.4 list-files.py . . . 22
- 4.5 mac-to-eui64.py . . . 22
- 4.6 pdf-compress.py . . . 23
- 4.7 image-into-pdf.py . . . 23

Python Module Index **25**

Index **27**

Contents:

CHAPTER 1

Comande line interfaces

1.1 dns-ipv6-prefix.py

Get the ipv6 prefix from a DNS name.

```
usage: dns-ipv6-prefix.py [-h] [-V] dnsname
```

1.1.1 Positional Arguments

dnsname The DNS name, e. g. josef-friedrich.de

1.1.2 Named Arguments

-V, --version show program's version number and exit

1.2 extract-pdftext.py

```
usage: extract-pdftext.py [-h] [-c] [-v] [-V] file
```

1.2.1 Positional Arguments

file A PDF file containing text

1.2.2 Named Arguments

-c, --colorize Colorize the terminal output.

Default: False

-v, --verbose Make the command line output more verbose.

Default: False

-V, --version show program's version number and exit

1.3 find-dupes-by-size.py

Find duplicate files by size.

```
usage: find-dupes-by-size.py [-h] [-V] path
```

1.3.1 Positional Arguments

path A directory to recursively search for duplicate files.

1.3.2 Named Arguments

-V, --version show program's version number and exit

1.4 list-files.py

This is a script to demonstrate the list_files() function in this file.

list-files.py a.txt list-files.py a.txt b.txt c.txt list-files.py (asterisk).txt list-files.py "(asterisk).txt" list-files.py dir/ list-files.py "dir/(asterisk).txt"

```
usage: list-files.py [-h] [-V] input_files [input_files ...]
```

1.4.1 Positional Arguments

input_files Examples for this arguments are: "a.txt", "a.txt b.txt c.txt", "(asterisk).txt", ""(asterisk).txt"", "dir/", ""dir/(asterisk).txt""

1.4.2 Named Arguments

-V, --version show program's version number and exit

1.5 mac-to-eui64.py

Convert mac addresses to EUI64 ipv6 addresses.

```
usage: mac-to-eui64.py [-h] [-V] mac prefix
```

1.5.1 Positional Arguments

mac The mac address.

prefix The ipv6 /64 prefix.

1.5.2 Named Arguments

-V, --version show program's version number and exit

1.6 pdf-compress.py

Convert and compress PDF scans. Make scans suitable for imslp.org (International Music Score Library Project). See also http://imslp.org/wiki/IMSLP:Musiknoten_beisteuern. The output files are monochrome bitmap images at a resolution of 600 dpi and the compression format CCITT group 4.

```
usage: pdf-compress.py [-h] [-c] [-m] [-N] [-v] [-V]
                       {convert,con,c,extract,ex,e,join,jn,j,samples,sp,s,unify,un,u}
                       ...
```

1.6.1 Positional Arguments

subcommand Possible choices: convert, con, c, extract, ex, e, join, jn, j, samples, sp, s, unify, un, u

Subcommand

1.6.2 Named Arguments

-c, --colorize Colorize the terminal output.

Default: False

-m, --multiprocessing Use multiprocessing to run commands in parallel.

Default: False

-N, --no-cleanup Don't clean up the temporary files.

Default: False

-v, --verbose Make the command line output more verbose.

Default: False

-V, --version show program's version number and exit

1.6.3 Sub-commands:

convert (con, c)

Convert scanned images (can be many image file formats or a PDF files) in monochrome bitmap images. The resulting images are compressed using the CCITT group 4 compression.

```
pdf-compress.py convert [-h] [-a | -C | -P] [-b] [--blur BLUR] [-B] [-c] [-d]
                        [-e] [-f] [-j] [-o]
                        [-l OCR_LANGUAGE [OCR_LANGUAGE ...]] [-p] [-n]
                        [-q QUALITY] [-r] [-t THRESHOLD] [-T] [-u]
                        input_files [input_files ...]
```

Positional Arguments

input_files a.tiff a.tiff b.tiff c.tiff (asterisk).tiff "(asterisk).tiff" dir/ "dir/(asterisk).tiff"

Named Arguments

-a, --auto-black-white The same as "–deskew –join –ocr –pdf –resize –trim –unify"

Default: False

-C, --auto-color The same as "–color –deskew –join –ocr –pdf –resize –trim –unify"

Default: False

-P, --auto-png The same as "–deskew –resize –trim"

Default: False

-b, --backup Backup original images (add _backup.ext to filename).

Default: False

--blur Blur images for better jpeg2000 compression rate.

Default: False

-B, --border Frame the images with a white border.

Default: False

-c, --color The input files are colored images.

Default: False

-d, --deskew Straighten the images.

Default: False

-e, --enlighten-border Enlighten the border.

Default: False

-f, --force Overwrite the output file even if it exists and it seems to be already converted.

Default: False

-j, --join	Join single paged PDF files to one PDF file. This option takes only effect with the option –pdf. Default: False
-o, --ocr	Perform optical character recognition (OCR) on the input files.The output format must be PDF. Default: False
-l, --ocr-language	Run tesseract –list-langs to get your installed languages.
-p, --pdf	Generate a PDF file. Default: False
-n, --png	Generate a PNG file. Default: False
-q, --quality	Compress the input images in a specific quality. The command automatically turns into the color mode. Default: False
-r, --resize	Resize 200 percent. Default: False
-t, --threshold	Threshold for monochrome, black and white images, default 50 percent. Colors above the threshold will be white and below will be black. Default: 50%
-T, --trim	This option removes any edges that are exactly the same color as the corner pixels. Default: False
-u, --unify	Unify the page size of all pages in a PDF File. The output must be a joined PDF. Default: False

extract (ex, e)

Extract images from a PDF file and export them in the TIFF format.

```
pdf-compress.py extract [-h] input_file [input_file ...]
```

Positional Arguments

input_file	A pdf file

join (jn, j)

Join the input files into a single PDF file. If the input file is not PDF file, it is converted into a monochrome CCITT Group 4 compressed PDF file.

```
pdf-compress.py join [-h] [-o] [-l OCR_LANGUAGE [OCR_LANGUAGE ...]]
                     input_files [input_files ...]
```

Positional Arguments

input_files a.png a.png b.png c.png (asterisk).png "(asterisk).png" dir/ "dir/(asterisk).png"

Named Arguments

-o, --ocr Perform optical character recognition (OCR) on the input files.

Default: False

-l, --ocr-language Run tesseract –list-langs to get your installed languages.

samples (sp, s)

Convert the samge image with different threshold values to find the best threshold value.

```
pdf-compress.py samples [-h] [-b] [-q] [-t] input_file
```

Positional Arguments

input_file A image or a PDF file. The script selects randomly one page of a multipaged PDF to build the series with differnt threshold values.

Named Arguments

-b, --blur Convert images on different blur values.

Default: False

-q, --quality Compress to JPEG2000 images in different quality steps.

Default: False

-t, --threshold Convert images on different threshold values to monochrome black and white images.

Default: False

unify (un, u)

Unify the page size of all pages in a PDF File.

```
pdf-compress.py unify [-h] [-m MARGIN] input_file
```

Positional Arguments

input_file A PDF file

Named Arguments

-m, --margin Add a margin around each page in the PDF file.

1.7 image-into-pdf.py

Add or replace one page in a PDF file with an image file of the same page size.

```
usage: image-into-pdf.py [-h] [-c] [-v] [-V]
                         {add,ad,a,convert,cv,c,replace,re,r} ...
```

1.7.1 Positional Arguments

subcmd_args Possible choices: add, ad, a, convert, cv, c, replace, re, r

Subcmd_args

1.7.2 Named Arguments

-c, --colorize Colorize the terminal output.

Default: False

-v, --verbose Make the cmd_args line output more verbose.

Default: False

-V, --version show program's version number and exit

1.7.3 Sub-commands:

add (ad, a)

Add one image to an PDF file.

```
image-into-pdf.py add [-h] [-a AFTER | -b BEFORE | -f | -l] image pdf
```

Positional Arguments

image A image (or a PDF) file to add to the PDF page.

pdf The PDF file.

Named Arguments

-a, --after Place image after page X.

-b, --before Place image before page X.

-f, --first	Place the image to the first position. Default: False
-l, --last	Place the image to the last position. Default: False

convert (cv, c)

Convert a image file into a PDF file with the same dimensions.

```
image-into-pdf.py convert [-h] image pdf
```

Positional Arguments

image	The image file to convert to the PDF format.
pdf	The main PDF file (to get the dimensions).

replace (re, r)

Replace one page in a PDF file with an image (or an PDF) file.

```
image-into-pdf.py replace [-h] pdf number image
```

Positional Arguments

pdf	The main PDF file
number	The page number of the PDF page to replace.
image	A image (or a PDF) file to replace the PDF page with.

CHAPTER 2

jfscripts

2.1 jfscripts package

2.1.1 Submodules

jfscripts._utils module

class jfscripts._utils.**FilePath**(*path*, *absolute=False*)

Bases: object

_export(*path*)

absolute = None

Boolean, indicates wheter the path is an absolute path or an relative path.

base = None

The path without an extension, e. g. */home/document/file*.

basename = None

The basename of the file, e. g. *file*.

extension = None

The extension of the file, e. g. *ext*.

filename = None

The filename is the combination of the basename and the extension, e. g. *file.ext*.

new(*extension=None*, *append=''*, *del_substring=''*)

Parameters

- **extension** (*str*) – The extension of the new file path.
- **append** (*str*) – String to append on the basename. This string is located before the extension.
- **del_substring** (*str*) – String to delete from the new file path.

Returns A new file path object.

Return type *FilePath*

path = None
The absolute (*/home/document/file.ext*) or the relative path (*document/file.ext*) of the file.

remove()
Remove the file.

class jfscripts._utils.**Run**(**args*, ***kwargs*)
Bases: object

PIPE = -1

_print_cmd(*cmd*)

check_output(**args*, ***kwargs*)

run(**args*, ***kwargs*)

Returns A *CompletedProcess* object.

Return type subprocess.CompletedProcess

setup(*verbose=False*, *colorize=False*)

jfscripts._utils.**argparser_to_readme**(*argparser*, *template='README-template.md'*, *destination='README.md'*, *indentation=0*, *placeholder='{{ argparse }}'*)
Add the formatted help output of a command line utility using the Python module *argparse* to a README file. Make sure to set the name of the program (*prop*) or you get strange program names.

Parameters

- **argparser** (*object*) – The argparse parser object.
- **template** (*str*) – The path of a template text file containing the placeholder. Default: *README-template.md*
- **destination** (*str*) – The path of the destination file. Default: *README.me*
- **indentation** (*int*) – Indent the formatted help output by X spaces. Default: 0
- **placeholder** (*str*) – Placeholder string that gets replaced by the formatted help output. Default: *{{ argparse }}*

jfscripts._utils.**check_dependencies**(**executables*, *raise_error=True*)
Check if the given executables are existing in $PATH.

Parameters

- **executables** (*tuple*) – A tuple of executables to check for their existence in $PATH. Each element of the tuple can be either a string (e. g. *pdfimages*) or a itself a tuple *('pdfimages', 'poppler')*. The first entry of this tuple is the name of the executable the second entry is a description text which is displayed in the raised exception.
- **raise_error** (*bool*) – Raise an error if an executable doesn't exist.

Returns True or False. True if all executables exist. False if one or more executables not exist.

Return type bool

jfscripts.dns_ipv6_prefix module

jfscripts.dns_ipv6_prefix.**get_ipv6**(*dns_name*)

jfscripts.dns_ipv6_prefix.**get_parser**()
The argument parser for the command line interface.

Returns A ArgumentParser object.

Return type argparse.ArgumentParser

jfscripts.dns_ipv6_prefix.**main**()

jfscripts.extract_pdftext module

class jfscripts.extract_pdftext.**Txt**(*path*)
Bases: object

add_line(*line*)

jfscripts.extract_pdftext.**get_page_count**(*pdf*)

jfscripts.extract_pdftext.**get_parser**()
The argument parser for the command line interface.

Returns A ArgumentParser object.

Return type argparse.ArgumentParser

jfscripts.extract_pdftext.**get_text_per_page**(*pdf*, *page*, *txt_file*)

jfscripts.extract_pdftext.**main**()

jfscripts.find_dupes_by_size module

jfscripts.find_dupes_by_size.**check_for_duplicates**(*path*)

jfscripts.find_dupes_by_size.**get_parser**()
The argument parser for the command line interface.

Returns A ArgumentParser object.

Return type argparse.ArgumentParser

jfscripts.find_dupes_by_size.**main**()

jfscripts.list_files module

jfscripts.list_files.**_list_files_all**(*dir_path*)

jfscripts.list_files.**_list_files_filter**(*dir_path*, *glob_pattern*)

jfscripts.list_files.**_split_glob**(*glob_path*)
Split a file path (e. g.: /data/(asterisk).txt) containing glob wildcard characters in a glob free path prefix (e. g.: /data) and a glob pattern (e. g. (asterisk).txt).

Parameters **glob_path** (*str*) – A file path containing glob wildcard characters.

jfscripts.list_files.**common_path**(*paths*)

jfscripts.list_files.**doc_examples**(*command_name=''*, *extension='txt'*, *indent_spaces=0*, *inline=False*)

jfscripts.list_files.**get_parser**()
The argument parser for the command line interface.

Returns A ArgumentParser object.

Return type argparse.ArgumentParser

jfscripts.list_files.**is_glob**(*string*)

jfscripts.list_files.**list_files**(*files*, *default_glob=None*)

Parameters

- **files** (*list*) – A list of file paths or a single element list containing a glob string.
- **default_glob** (*string*) – A default glob pattern like "(asterisk).txt". This argument is only taken into account, if "element" is a list with only one entry and this entry is a path to a directory.

jfscripts.list_files.**main**()

jfscripts.mac_to_eui64 module

jfscripts.mac_to_eui64.**get_parser**()
The argument parser for the command line interface.

Returns A ArgumentParser object.

Return type argparse.ArgumentParser

jfscripts.mac_to_eui64.**mac_to_eui64**(*mac*, *prefix=None*)
Convert a MAC address to a EUI64 address or, with prefix provided, a full IPv6 address

jfscripts.mac_to_eui64.**main**()

jfscripts.pdf_compress module

class jfscripts.pdf_compress.**State**(*args*)
Bases: object

This object holds runtime data for the multiprocessing environment.

args = None
argparse arguments

common_path = None
The common path prefix of all input files.

cwd = None
The current working directory

first_input_file = None
The first input file.

input_files = None
A list of all input files.

input_is_pdf = None
Boolean that indicates if the first file is a pdf.

class jfscripts.pdf_compress.**Timer**
Bases: object

Class to calculate the execution time. Mainly to test the speed improvements of the multiprocessing implementation.

begin = None
UNIX timestamp the execution began.

end = None
UNIX timestamp the execution ended.

stop()
Stop the time calculation and return the formated result.

Returns The result

Return type str

jfscripts.pdf_compress.**_do_magick_command**(*command*)
ImageMagick version 7 introduces a new top level command named *magick*. Use this newer command if present.

Returns A list of command segments

jfscripts.pdf_compress.**_do_magick_convert_enlighten_border**(*width*, *height*)
Build the command line arguments to enlighten the border in four regions.

Parameters

- **width** (*int*) – The width of the image.
- **height** (*int*) – The height of the image.

Returns Command line arguments for imagemagicks' *convert*.

Return type list

jfscripts.pdf_compress.**args = None**
The argparse object.

jfscripts.pdf_compress.**check_threshold**(*value*)
Check if *value* is a valid threshold value.

Parameters **value** (*integer or string*) –

Returns A normalized threshold string (*90%*)

Return type string

jfscripts.pdf_compress.**cleanup**(*state*)
Delete all images using the temporary identifier in a common path.

Parameters **state** (jfscripts.pdf_compress.State) – The state object.

Returns None

jfscripts.pdf_compress.**collect_images**(*state*)
Collection all images using the temporary identifier in a common path.

Parameters **state** (jfscripts.pdf_compress.State) – The state object.

Returns A sorted list of image paths.

Return type list

jfscripts.pdf_compress.**convert_file_paths**(*files*)
Convert a list of file paths in a list of *jfscripts._utils.FilePath* objects.

Parameters **files** (*list*) – A list of file paths

Returns a list of *jfscripts._utils.FilePath* objects.

jfscripts.pdf_compress.**do_magick_convert**(*input_file*, *output_file*, *threshold=None*, *enlighten_border=False*, *border=False*, *resize=False*, *deskew=False*, *trim=False*, *color=False*, *quality=75*, *blur=False*)

Convert a input image file using the subcommand convert of the imagemagick suite.

Returns The output image file.

Return type *jfscripts._utils.FilePath*

jfscripts.pdf_compress.**do_magick_identify**(*input_file*)

The different informations of an image.

Parameters **input_file** (jfscripts._utils.FilePath) – The input file.

Returns A directory with the keys *width*, *height* and *colors*.

Return type dict

jfscripts.pdf_compress.**do_pdfimages**(*pdf_file*, *state*, *page_number=None*, *use_tmp_identifier=True*)

Convert a PDF file to images in the TIFF format.

Parameters

- **pdf_file** (jfscripts._utils.FilePath) – The input file.
- **state** (jfscripts.pdf_compress.State) – The state object.
- **page_number** (*int*) – Extract only the page with a specific page number.

Returns The return value of *subprocess.run*.

Return type subprocess.CompletedProcess

jfscripts.pdf_compress.**do_pdfinfo_page_count**(*pdf_file*)

Get the amount of pages a PDF files have.

Parameters **pdf_file** (*str*) – Path of the PDF file.

Returns Page count

Return type int

jfscripts.pdf_compress.**do_pdftk_cat**(*pdf_files*, *state*)

Join a list of PDF files into a single PDF file using the tool *pdftk*.

Parameters

- **pdf_files** (*list*) – a list of PDF files
- **state** (jfscripts.pdf_compress.State) – The state object.

Returns None

jfscripts.pdf_compress.**do_tesseract**(*input_file, languages=['deu', 'eng']*)

jfscripts.pdf_compress.**get_parser**()

The argument parser for the command line interface.

Returns A ArgumentParser object.

Return type argparse.ArgumentParser

jfscripts.pdf_compress.**identifier = 'magick'**
To allow better assignment of the output files.

jfscripts.pdf_compress.**main**()
Main function.

Returns None

jfscripts.pdf_compress.**state = None**
The global *State* object.

jfscripts.pdf_compress.**subcommand_convert_file**(*arguments*)
Manipulate one input file

Parameters **arguments** (*tuple*) – A tuple containing two elements: The first element is the input_file file object and the second element is the state object.

jfscripts.pdf_compress.**subcommand_join_convert_pdf**(*arguments*)

jfscripts.pdf_compress.**subcommand_samples**(*input_file*, *state*)
Generate a list of example files with different threshold values.

Parameters

- **input_file** (jfscripts._utils.FilePath) – The input file.
- **state** (jfscripts.pdf_compress.State) – The state object.

Returns None

jfscripts.pdf_compress.**tmp_identifier = 'magick_9dd9904a-0159-11ed-8507-0242ac110002'**
Used for the identification of temporary files.

jfscripts.pdf_compress.**unify_page_size**(*input_file*, *output_file*, *margin=0*)

jfscripts.image_into_pdf module

jfscripts.image_into_pdf.**assemble_pdf**(*main_pdf*, *insert_pdf*, *page_count*, *page_number*, *mode='add'*, *position='before'*)

Parameters

- **main_pdf** (*str*) – Path of the main PDF file.
- **insert_pdf** (*str*) – Path of the PDF file to insert into the main PDF file.
- **page_count** (*int*) – Page count of the main PDF file.
- **page_number** (*int*) – Page number in the main PDF file to add / to replace the insert PDF file.
- **mode** (*string*) – Mode how the PDF to insert is treated. Possible choices are: *add* or *replace*.
- **position** (*str*) – Possible choices: *before* and *after*

jfscripts.image_into_pdf.**convert_image_to_pdf_page**(*image*, *image_width*, *pdf_width*, *pdf_density_x*)

jfscripts.image_into_pdf.**do_magick_identify_dimensions**(*pdf_file*)

jfscripts.image_into_pdf.**do_pdftk_cat_first_page**(*pdf_file*)
The cmd_args magick identify is very slow on page pages hence it examines every page. We extract the first page to get some informations about the dimensions of the PDF file.

jfscripts.image_into_pdf.**get_parser**()
The argument parser for the cmd_args line interface.

Returns A ArgumentParser object.

Return type argparse.ArgumentParser

jfscripts.image_into_pdf.**get_pdf_info**(*pdf_file*)

jfscripts.image_into_pdf.**main**()

CHAPTER 3

Indices and tables

- genindex
- modindex
- search

CHAPTER 4

jfscripts

A collection of my personal Python scripts.

4.1 dns-ipv6-prefix.py

```
usage: dns-ipv6-prefix.py [-h] [-V] dnsname

Get the ipv6 prefix from a DNS name.

positional arguments:
  dnsname        The DNS name, e. g. josef-friedrich.de

optional arguments:
  -h, --help     show this help message and exit
  -V, --version  show program's version number and exit
```

4.2 extract-pdftext.py

```
usage: extract-pdftext.py [-h] [-c] [-v] [-V] file

positional arguments:
  file            A PDF file containing text

optional arguments:
  -h, --help      show this help message and exit
  -c, --colorize  Colorize the terminal output.
  -v, --verbose   Make the command line output more verbose.
  -V, --version   show program's version number and exit
```

4.3 find-dupes-by-size.py

```
usage: find-dupes-by-size.py [-h] [-V] path

Find duplicate files by size.

positional arguments:
  path           A directory to recursively search for duplicate files.

optional arguments:
  -h, --help     show this help message and exit
  -V, --version  show program's version number and exit
```

4.4 list-files.py

```
usage: list-files.py [-h] [-V] input_files [input_files ...]

This is a script to demonstrate the list_files() function in this file.

list-files.py a.txt
list-files.py a.txt b.txt c.txt
list-files.py (asterisk).txt
list-files.py "(asterisk).txt"
list-files.py dir/
list-files.py "dir/(asterisk).txt"

positional arguments:
  input_files    Examples for this arguments are: "a.txt", "a.txt b.txt
                 c.txt", "(asterisk).txt", ""(asterisk).txt"", "dir/",
                 ""dir/(asterisk).txt""

optional arguments:
  -h, --help     show this help message and exit
  -V, --version  show program's version number and exit
```

4.5 mac-to-eui64.py

```
usage: mac-to-eui64.py [-h] [-V] mac prefix

Convert mac addresses to EUI64 ipv6 addresses.

positional arguments:
  mac            The mac address.
  prefix         The ipv6 /64 prefix.

optional arguments:
  -h, --help     show this help message and exit
  -V, --version  show program's version number and exit
```

4.6 pdf-compress.py

```
usage: pdf-compress.py [-h] [-c] [-m] [-N] [-v] [-V]
                       {convert,con,c,extract,ex,e,join,jn,j,samples,sp,s,unify,un,u}
                       ...

Convert and compress PDF scans. Make scans suitable for imslp.org
(International Music Score Library Project). See also
http://imslp.org/wiki/IMSLP:Musiknoten_beisteuern. The output files are
monochrome bitmap images at a resolution of 600 dpi and the compression format
CCITT group 4.

positional arguments:
  {convert,con,c,extract,ex,e,join,jn,j,samples,sp,s,unify,un,u}
                        Subcommand

optional arguments:
  -h, --help            show this help message and exit
  -c, --colorize        Colorize the terminal output.
  -m, --multiprocessing
                        Use multiprocessing to run commands in parallel.
  -N, --no-cleanup      Don't clean up the temporary files.
  -v, --verbose         Make the command line output more verbose.
  -V, --version         show program's version number and exit
```

4.7 image-into-pdf.py

```
usage: image-into-pdf.py [-h] [-c] [-v] [-V]
                         {add,ad,a,convert,cv,c,replace,re,r} ...

Add or replace one page in a PDF file with an image file of the same page
size.

positional arguments:
  {add,ad,a,convert,cv,c,replace,re,r}
                        Subcmd_args

optional arguments:
  -h, --help            show this help message and exit
  -c, --colorize        Colorize the terminal output.
  -v, --verbose         Make the cmd_args line output more verbose.
  -V, --version         show program's version number and exit
```

Python Module Index

j

jfscripts._utils, 11
jfscripts.dns_ipv6_prefix, 13
jfscripts.extract_pdftext, 13
jfscripts.find_dupes_by_size, 13
jfscripts.image_into_pdf, 17
jfscripts.list_files, 13
jfscripts.mac_to_eui64, 14
jfscripts.pdf_compress, 14

Index

Symbols

_do_magick_command() (*in module jfscripts.pdf_compress*), 15
_do_magick_convert_enlighten_border() (*in module jfscripts.pdf_compress*), 15
_export() (*jfscripts._utils.FilePath method*), 11
_list_files_all() (*in module jfscripts.list_files*), 13
_list_files_filter() (*in module jfscripts.list_files*), 13
_print_cmd() (*jfscripts._utils.Run method*), 12
_split_glob() (*in module jfscripts.list_files*), 13

A

absolute (*jfscripts._utils.FilePath attribute*), 11
add_line() (*jfscripts.extract_pdftext.Txt method*), 13
argparser_to_readme() (*in module jfscripts._utils*), 12
args (*in module jfscripts.pdf_compress*), 15
args (*jfscripts.pdf_compress.State attribute*), 14
assemble_pdf() (*in module jfscripts.image_into_pdf*), 17

B

base (*jfscripts._utils.FilePath attribute*), 11
basename (*jfscripts._utils.FilePath attribute*), 11
begin (*jfscripts.pdf_compress.Timer attribute*), 15

C

check_dependencies() (*in module jfscripts._utils*), 12
check_for_duplicates() (*in module jfscripts.find_dupes_by_size*), 13
check_output() (*jfscripts._utils.Run method*), 12
check_threshold() (*in module jfscripts.pdf_compress*), 15
cleanup() (*in module jfscripts.pdf_compress*), 15
collect_images() (*in module jfscripts.pdf_compress*), 15
common_path (*jfscripts.pdf_compress.State attribute*), 14
common_path() (*in module jfscripts.list_files*), 13
convert_file_paths() (*in module jfscripts.pdf_compress*), 15
convert_image_to_pdf_page() (*in module jfscripts.image_into_pdf*), 17
cwd (*jfscripts.pdf_compress.State attribute*), 14

D

do_magick_convert() (*in module jfscripts.pdf_compress*), 16
do_magick_identify() (*in module jfscripts.pdf_compress*), 16
do_magick_identify_dimensions() (*in module jfscripts.image_into_pdf*), 17
do_pdfimages() (*in module jfscripts.pdf_compress*), 16
do_pdfinfo_page_count() (*in module jfscripts.pdf_compress*), 16
do_pdftk_cat() (*in module jfscripts.pdf_compress*), 16
do_pdftk_cat_first_page() (*in module jfscripts.image_into_pdf*), 17
do_tesseract() (*in module jfscripts.pdf_compress*), 16
doc_examples() (*in module jfscripts.list_files*), 13

E

end (*jfscripts.pdf_compress.Timer attribute*), 15
extension (*jfscripts._utils.FilePath attribute*), 11

F

filename (*jfscripts._utils.FilePath attribute*), 11
FilePath (*class in jfscripts._utils*), 11
first_input_file (*jfscripts.pdf_compress.State attribute*), 14

G

get_ipv6() (*in module jfscripts.dns_ipv6_prefix*), 13

get_page_count() (*in module jfscripts.extract_pdftext*), 13
get_parser() (*in module jfscripts.dns_ipv6_prefix*), 13
get_parser() (*in module jfscripts.extract_pdftext*), 13
get_parser() (*in module jfscripts.find_dupes_by_size*), 13
get_parser() (*in module jfscripts.image_into_pdf*), 17
get_parser() (*in module jfscripts.list_files*), 13
get_parser() (*in module jfscripts.mac_to_eui64*), 14
get_parser() (*in module jfscripts.pdf_compress*), 16
get_pdf_info() (*in module jfscripts.image_into_pdf*), 18
get_text_per_page() (*in module jfscripts.extract_pdftext*), 13

I

identifier (*in module jfscripts.pdf_compress*), 16
input_files (*jfscripts.pdf_compress.State attribute*), 14
input_is_pdf (*jfscripts.pdf_compress.State attribute*), 14
is_glob() (*in module jfscripts.list_files*), 14

J

jfscripts._utils (*module*), 11
jfscripts.dns_ipv6_prefix (*module*), 13
jfscripts.extract_pdftext (*module*), 13
jfscripts.find_dupes_by_size (*module*), 13
jfscripts.image_into_pdf (*module*), 17
jfscripts.list_files (*module*), 13
jfscripts.mac_to_eui64 (*module*), 14
jfscripts.pdf_compress (*module*), 14

L

list_files() (*in module jfscripts.list_files*), 14

M

mac_to_eui64() (*in module jfscripts.mac_to_eui64*), 14
main() (*in module jfscripts.dns_ipv6_prefix*), 13
main() (*in module jfscripts.extract_pdftext*), 13
main() (*in module jfscripts.find_dupes_by_size*), 13
main() (*in module jfscripts.image_into_pdf*), 18
main() (*in module jfscripts.list_files*), 14
main() (*in module jfscripts.mac_to_eui64*), 14
main() (*in module jfscripts.pdf_compress*), 17

N

new() (*jfscripts._utils.FilePath method*), 11

P

path (*jfscripts._utils.FilePath attribute*), 12
PIPE (*jfscripts._utils.Run attribute*), 12

R

remove() (*jfscripts._utils.FilePath method*), 12
Run (*class in jfscripts._utils*), 12
run() (*jfscripts._utils.Run method*), 12

S

setup() (*jfscripts._utils.Run method*), 12
State (*class in jfscripts.pdf_compress*), 14
state (*in module jfscripts.pdf_compress*), 17
stop() (*jfscripts.pdf_compress.Timer method*), 15
subcommand_convert_file() (*in module jfscripts.pdf_compress*), 17
subcommand_join_convert_pdf() (*in module jfscripts.pdf_compress*), 17
subcommand_samples() (*in module jfscripts.pdf_compress*), 17

T

Timer (*class in jfscripts.pdf_compress*), 14
tmp_identifier (*in module jfscripts.pdf_compress*), 17
Txt (*class in jfscripts.extract_pdftext*), 13

U

unify_page_size() (*in module jfscripts.pdf_compress*), 17