PDF4Cat

Release 0.4.2

blackcatdev

Jun 24, 2022

CONTENTS

CHAPTER

ONE

PDF4CAT MODULE

1.1 PDF4Cat.Converter class

1.1.1 Class bases

class PDF4Cat.converter.**Converter**(**args*, ***kwargs*)

Bases: *Img2Pdf*, *Pdf2Img*, *OCR*, *any_doc_convert*, *soffice_convert*

Parent class of PDF4Cat.converter submodule

class PDF4Cat.converter.**any_doc_convert**(**args*, ***kwargs*)

Bases: PDF4Cat

Subclass of PDF4Cat parent class

Parameters

- **doc_file** (*None, optional*) – Document file (for multiple operations, 'use input_doc_list')
- **input_doc_list** (*list, optional*) – List of input docs
- **passwd** (*str, optional*) – Document password (for crypt/decrypt)
- **progress_callback** (*None, optional*) – Progress callback like:

Raises

TypeError – If you use doc_file with input_doc_list (you can use only one)

convert2pdf(*output_pdf*, *use_soffice=False*)

Pdf to any (using PyMuPDF or Libre Office)

Parameters

- **output_pdf** (*None, optional*) – Output pdf file
- **use_soffice** (*bool, optional*) – Use Libre Office converter

docx2html(*output_doc*, *style_map=None*)

docx to html (using PyMuPDF)

Parameters

output_html (*None, optional*) – Output html file

docx2pdf(*output_pdf*)

docx to pdf (using PyMuPDF [docx=>html=>pdf])

Parameters
output_pdf (*None, optional*) – Output pdf file

gen_images4conv(*pdf*) → bytes

Generator, generate BytesIO object

Parameters
pdf (*None, optional*) – pdf object (PDF4Cat.open)

Yields
bytes – BytesIO

pdf2docx(*output_docx*)

Pdf to docx (using PyMuPDF)

Parameters
output_docx (*None, optional*) – Output docx file

pdf2pptx(*output_pptx*, *A4=True*)

Pdf to pptx (using PyMuPDF)

Parameters

- **output_pptx** (*None, optional*) – Output pptx file
- **A4** (*bool, optional*) – Use Inches for A4 page

class PDF4Cat.converter.**Img2Pdf**(**args*, ***kwargs*)

Bases: PDF4Cat

Subclass of PDF4Cat parent class

Parameters

- **doc_file** (*None, optional*) – Document file (for multiple operations, 'use input_doc_list')
- **input_doc_list** (*list, optional*) – List of input docs
- **passwd** (*str, optional*) – Document password (for crypt/decrypt)
- **progress_callback** (*None, optional*) – Progress callback like:

Raises
TypeError – If you use doc_file with input_doc_list (you can use only one)

gen_imagesi2p(*fimages: str = '{name}_{num}.pdf'*, *start_from: int = 0*) → tuple

Generator, generate name with BytesIO object

Parameters

- **fimages** (*str, optional*) – Format image filenames
- **start_from** (*int, optional*) – Enumerate from n

Yields
tuple – filename, BytesIO

img2pdf(*output_pdf=None*) → None

Image to pdf

Parameters
output_pdf (*None, optional*) – Output pdf file

imgs2pdf(*output_pdf=None*) → None

Multiple images to pdf

Parameters

output_pdf (*None, optional*) – Output pdf file

imgs2pdfs_zip(*out_zip_file: str*, *fimages: str = '{name}_{num}.pdf'*, *start_from: int = 0*) → None

Multiple images to multiple pdfs and compress to zip (using gen_imagesi2p generator)

Parameters

- **out_zip_file** (*str*) – Output zip file
- **fimages** (*str, optional*) – Format image filenames
- **start_from** (*int, optional*) – Enumerate from n

class PDF4Cat.converter.**Pdf2Img**(**args*, ***kwargs*)

Bases: PDF4Cat

Subclass of PDF4Cat parent class

Parameters

- **doc_file** (*None, optional*) – Document file (for multiple operations, 'use input_doc_list')
- **input_doc_list** (*list, optional*) – List of input docs
- **passwd** (*str, optional*) – Document password (for crypt/decrypt)
- **progress_callback** (*None, optional*) – Progress callback like:

Raises

TypeError – If you use doc_file with input_doc_list (you can use only one)

gen_imagesp2i(*pages: list = []*, *fimages: str = '{name}_{num}.png'*, *start_from: int = 0*, *zoom: float = 1.5*) → tuple

Generator, generate name with BytesIO object

Parameters

- **pages** (*list, optional*) – List of pages to select like [1, 3, 5, 15]
- **fimages** (*str, optional*) – Format image filenames
- **start_from** (*int, optional*) – Enumerate from n
- **zoom** (*float, optional*) – Zoom image (look fitz.Matrix docs)

Yields

tuple – filename, BytesIO

pdf2imgs_zip(*out_zip_file: str*, *pages: list = []*, *fimages: str = '{name}_{num}.png'*, *start_from: int = 0*, *zoom: float = 1.5*) → None

Multiple pdfs to multiple images and compress to zip (using gen_imagesp2i generator)

Parameters

- **out_zip_file** (*str*) – Output zip file
- **pages** (*list, optional*) – List of pages to select like [1, 3, 5, 15]
- **fimages** (*str, optional*) – Format image filenames
- **start_from** (*int, optional*) – Enumerate from n

- **zoom** (*float, optional*) – Zoom image (look fitz.Matrix docs)

class PDF4Cat.converter.**OCR**(**args*, ***kwargs*)

Bases: PDF4Cat

Subclass of PDF4Cat parent class

Parameters

- **doc_file** (*None, optional*) – Document file (for multiple operations, 'use input_doc_list')
- **input_doc_list** (*list, optional*) – List of input docs
- **passwd** (*str, optional*) – Document password (for crypt/decrypt)
- **progress_callback** (*None, optional*) – Progress callback like:

Raises

TypeError – If you use doc_file with input_doc_list (you can use only one)

gen_pdfImagesOCR(*pages: list = []*, *language: str = 'eng'*, *zoom: float = 1.5*) → tuple

Generator, generate BytesIO object

Parameters

- **pages** (*list, optional*) – List of pages to select like [1, 3, 5, 15]
- **language** (*str, optional*) – Language to ocr (look fitz.pdfocr_tobytes)
- **zoom** (*float, optional*) – Zoom image (look fitz.Matrix docs)

Yields

tuple – BytesIO

pdfocr(*language: str = 'eng'*, *output_pdf=None*, *pages: list = []*, *start_from: int = 0*, *zoom: float = 1.5*) → None

OCR pdf to file

Parameters

- **language** (*str, optional*) – Language to ocr (look fitz.pdfocr_tobytes)
- **output_pdf** (*None, optional*) – Output pdf file
- **pages** (*list, optional*) – List of pages to select like [1, 3, 5, 15]
- **start_from** (*int, optional*) – Enumerate from n
- **zoom** (*float, optional*) – Zoom image (look fitz.Matrix docs)

class PDF4Cat.converter.**any_doc_convert**(**args*, ***kwargs*)

Bases: PDF4Cat

Subclass of PDF4Cat parent class

Parameters

- **doc_file** (*None, optional*) – Document file (for multiple operations, 'use input_doc_list')
- **input_doc_list** (*list, optional*) – List of input docs
- **passwd** (*str, optional*) – Document password (for crypt/decrypt)
- **progress_callback** (*None, optional*) – Progress callback like:

Raises
TypeError – If you use doc_file with input_doc_list (you can use only one)

convert2pdf(*output_pdf*, *use_soffice=False*)

Pdf to any (using PyMuPDF or Libre Office)

Parameters

- **output_pdf** (*None, optional*) – Output pdf file
- **use_soffice** (*bool, optional*) – Use Libre Office converter

docx2html(*output_doc*, *style_map=None*)

docx to html (using PyMuPDF)

Parameters
output_html (*None, optional*) – Output html file

docx2pdf(*output_pdf*)

docx to pdf (using PyMuPDF [docx=>html=>pdf])

Parameters
output_pdf (*None, optional*) – Output pdf file

gen_images4conv(*pdf*) → bytes

Generator, generate BytesIO object

Parameters
pdf (*None, optional*) – pdf object (PDF4Cat.open)

Yields
bytes – BytesIO

pdf2docx(*output_docx*)

Pdf to docx (using PyMuPDF)

Parameters
output_docx (*None, optional*) – Output docx file

pdf2pptx(*output_pptx*, *A4=True*)

Pdf to pptx (using PyMuPDF)

Parameters

- **output_pptx** (*None, optional*) – Output pptx file
- **A4** (*bool, optional*) – Use Inches for A4 page

class PDF4Cat.converter.**soffice_convert**(**args*, ***kwargs*)

Bases: PDF4Cat

Subclass of PDF4Cat parent class

Parameters

- **doc_file** (*None, optional*) – Document file (for multiple operations, 'use input_doc_list')
- **input_doc_list** (*list, optional*) – List of input docs
- **passwd** (*str, optional*) – Document password (for crypt/decrypt)
- **progress_callback** (*None, optional*) – Progress callback like:

Raises
TypeError – If you use doc_file with input_doc_list (you can use only one)

soffice_convert2pdf(*output_pdf: str*)

Libre Office converter wrapper for convert document to pdf

Parameters
output_pdf (*str*) – Output pdf file

Raises
NotImplementedError – If Libre Office not support this conversion

soffice_convert2pdf_a(*a: int*, *output_pdf: str*)

Libre Office converter wrapper for convert document to pdf/a

Parameters

- **a** (*int*) – A type (0, 1) [0 - pdf 1.4; 1 - pdf/a]
- **output_pdf** (*str*) – Output pdf file

Raises
NotImplementedError – If Libre Office not support this conversion

soffice_convert_to(*doc_type: str*, *output_doc: str*)

Libre Office converter wrapper for convert document to any supported by soffice

Parameters

- **doc_type** (*str*) – Output document type to convert
- **output_doc** (*str*) – Output document file

1.2 PDF4Cat.doc class

1.2.1 Class bases

class PDF4Cat.doc.**Doc**(**args*, ***kwargs*)

Bases: *Merger*, *Splitter*, *Crypter*, *Effects*, *PdfOptimizer*

Parent class of PDF4Cat.doc submodule

class PDF4Cat.doc.**Merger**(**args*, ***kwargs*)

Bases: PDF4Cat

Subclass of PDF4Cat parent class

Parameters

- **doc_file** (*None, optional*) – Document file (for multiple operations, 'use input_doc_list')
- **input_doc_list** (*list, optional*) – List of input docs
- **passwd** (*str, optional*) – Document password (for crypt/decrypt)
- **progress_callback** (*None, optional*) – Progress callback like:

Raises
TypeError – If you use doc_file with input_doc_list (you can use only one)

merge_file_with(*input_pdf*, *output_pdf=None*) → None

Merge pdf with other pdf to new file

Parameters

- **input_pdf** (*str*) – File to merge with main document
- **output_pdf** (*None, optional*) – output_pdf (None, optional): Output pdf file

merge_files_to(*output_pdf=None*) → None

Merge pdfs with multiple pdfs to new file

Parameters

output_pdf (*None, optional*) – Output pdf file

class PDF4Cat.doc.**Splitter**(**args*, ***kwargs*)

Bases: PDF4Cat

Subclass of PDF4Cat parent class

Parameters

- **doc_file** (*None, optional*) – Document file (for multiple operations, 'use input_doc_list')
- **input_doc_list** (*list, optional*) – List of input docs
- **passwd** (*str, optional*) – Document password (for crypt/decrypt)
- **progress_callback** (*None, optional*) – Progress callback like:

Raises

TypeError – If you use doc_file with input_doc_list (you can use only one)

gen_split(*from_pdf=None*, *pages: list = []*, *fpages: str = '{name}_{num}.pdf'*, *start_from: int = 0*) → tuple

Generator, generate name with BytesIO object

Parameters

- **from_pdf** (*None, optional*) – pdf document name (default use main doc from class param)
- **pages** (*list, optional*) – List of pages to select like [1, 3, 5, 15]
- **fpages** (*str, optional*) – Format pdf filenames
- **start_from** (*int, optional*) – Enumerate from n

Yields

tuple – filename, BytesIO

split_pages2zip(*out_zip_file: str*, *pages: list = []*, *fpages: str = '{name}_{num}.pdf'*, *start_from: int = 0*) → None

Split pages to different pdfs and compress to zip

Parameters

- **out_zip_file** (*str*) – Output zip file
- **pages** (*list, optional*) – List of pages to select like [1, 3, 5, 15]
- **fpages** (*str, optional*) – Format pdf filenames
- **start_from** (*int, optional*) – Enumerate from n

class PDF4Cat.doc.**Crypter**(**args*, ***kwargs*)

Bases: PDF4Cat

Subclass of PDF4Cat parent class

Parameters

- **doc_file** (*None, optional*) – Document file (for multiple operations, 'use input_doc_list')
- **input_doc_list** (*list, optional*) – List of input docs
- **passwd** (*str, optional*) – Document password (for crypt/decrypt)
- **progress_callback** (*None, optional*) – Progress callback like:

Raises

TypeError – If you use doc_file with input_doc_list (you can use only one)

crypt_to(*user_passwd: str = None*, *owner_passwd: str = None*, *perm: dict = None*, *crypt_type: int = 5*, *output_pdf: str = None*) → None

Crypt pdf and save to file (don't forget give password in class parameter)

Parameters

- **user_passwd** (*str, optional*) – Pdf user password
- **owner_passwd** (*str, optional*) – Pdf owner password
- **perm** (*dict, optional*) – Permissions see past example - :perm
- **crypt_type** (*int, optional*) – Crypt type, default AES256 (PDF4Cat.PDF_ENCRYPT_AES_256)
- **output_pdf** (*None, optional*) – Output pdf file

Raises

TypeError – "Missing user and owner password!"

perm = int(PDF4Cat.PDF_PERM_ACCESSIBILITY

PDF4Cat.PDF_PERM_PRINT
PDF4Cat.PDF_PERM_COPY
PDF4Cat.PDF_PERM_ANNOTATE)

decrypt_to(*output_pdf=None*) → None

Decrypt pdf and save to file (don't forget give password in class parameter)

Parameters

output_pdf (*None, optional*) – Output pdf file

class PDF4Cat.doc.**Effects**(**args*, ***kwargs*)

Bases: Rotate

Parent class of PDF4Cat.Doc submodule

class PDF4Cat.doc.**PdfOptimizer**(**args*, ***kwargs*)

Bases: PDF4Cat

Subclass of PDF4Cat parent class

Parameters

- **doc_file** (*None, optional*) – Document file (for multiple operations, 'use input_doc_list')
- **input_doc_list** (*list, optional*) – List of input docs
- **passwd** (*str, optional*) – Document password (for crypt/decrypt)
- **progress_callback** (*None, optional*) – Progress callback like:

Raises

TypeError – If you use doc_file with input_doc_list (you can use only one)

DeFlate_to(*output_pdf=None*) → None

Deflate pdf to file

Parameters

output_pdf (*None, optional*) – Output pdf file

1.3 PDF4Cat.helpers module

PDF4Cat.helpers.**run_in_subprocess**(*func*)

A decorator adding a kwarg to a function that makes it run in a subprocess. This can be useful when you have a function that may segfault. You can use by call: @PDF4Cat.run_in_subprocess kwargs: run_in_subprocess=True, subprocess_timeout already using in: PDF4Cat.Converter funcs and PDF4Cat.Doc funcs

CHAPTER

TWO

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

p

INDEX