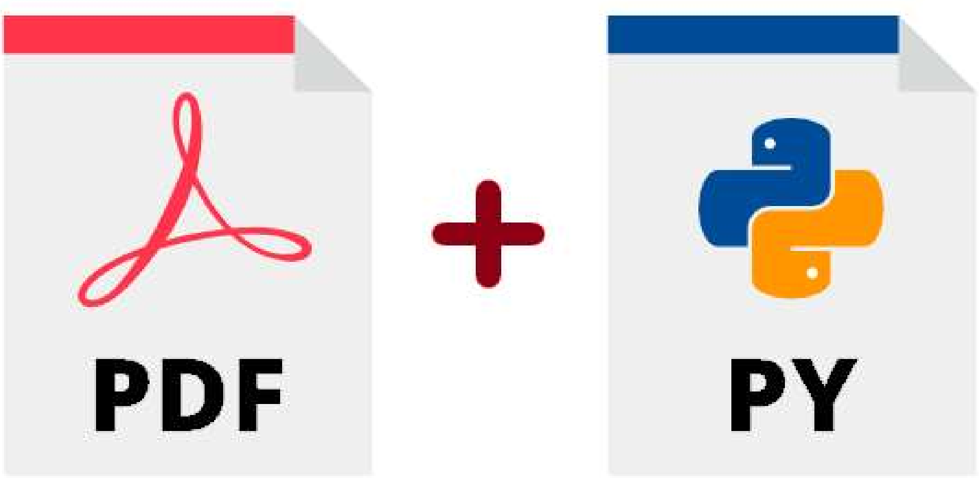
This is demo version!

Pdf in Python by PyPDF3



What is PyPDF3?

PyPDF3 is a pure-python library for working with PDF files.

We can use PyPDF3 to:

- **Extract document information from a PDF file in Python**
- **Rotate pages**
- **Merge PDFs**
- **Split PDFs**

- Encrypt a PDF

1. How to install and use the PyPDF3

In the pip write => **pip install PyPDF3**

Using PyPDF3 => `import PyPDF3 as MyPdf`

==

2. Extracting document information (Title, Author , …)

We can extract these information from Pdf file:

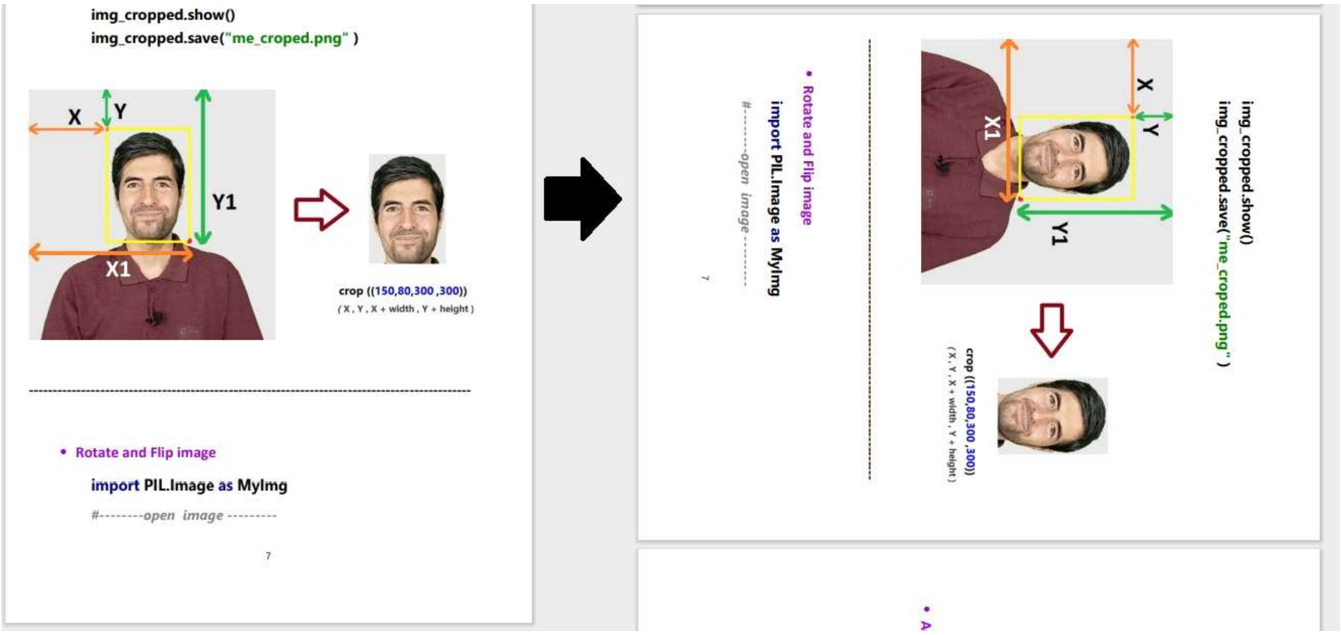
- **Number of pages**
- **Author**
- **Creator**
- **Producer**
- **Subject**
- **Title**
- **…**

```
import PyPDF3  as MyPdf
#--------- open pdf file & create pdf reader object ------
MyPdfFile =open("files/test.pdf","rb")
pdf_reader=MyPdf.PdfFileReader(MyPdfFile, strict=True )
#---------get data from pdf file -----
print(pdf_reader.getNumPages() )
doc_info=pdf_reader.getDocumentInfo()
print(doc_info)
#---------- Go through all data---------
for item in doc_info:
    print("Item: ",item)
    print("Value: ", doc_info[str(item)] )
    #------------
    FullInfo=str(item).removeprefix("/")+":"+doc_info[str(item)]
    print(FullInfo)
#-------------Get info one by one-------
DocAuthor=doc_info['/Author']
DocCreator=doc_info['/Creator']
DocProducer=doc_info['/Producer']
```

```
DocCreationDate=doc_info['/CreationDate']
#-------Combile all info -------
info=f"""
Author:{DocAuthor} _
Creator:{DocCreator} _
Producer:{DocProducer} _
Subject:{DocSubject} _
Title:{DocTitle} _
CreationDate:{DocCreationDate}
"""
#----------------------
print(info)
MyPdfFile.close()
```

==

2. Rotating PDF pages



```
img_cropped.show()
img_cropped.save("me_croped.png" )
```

crop ((150,80,300,300))
(X , Y , X + width , Y + height)

- Rotate and Flip image

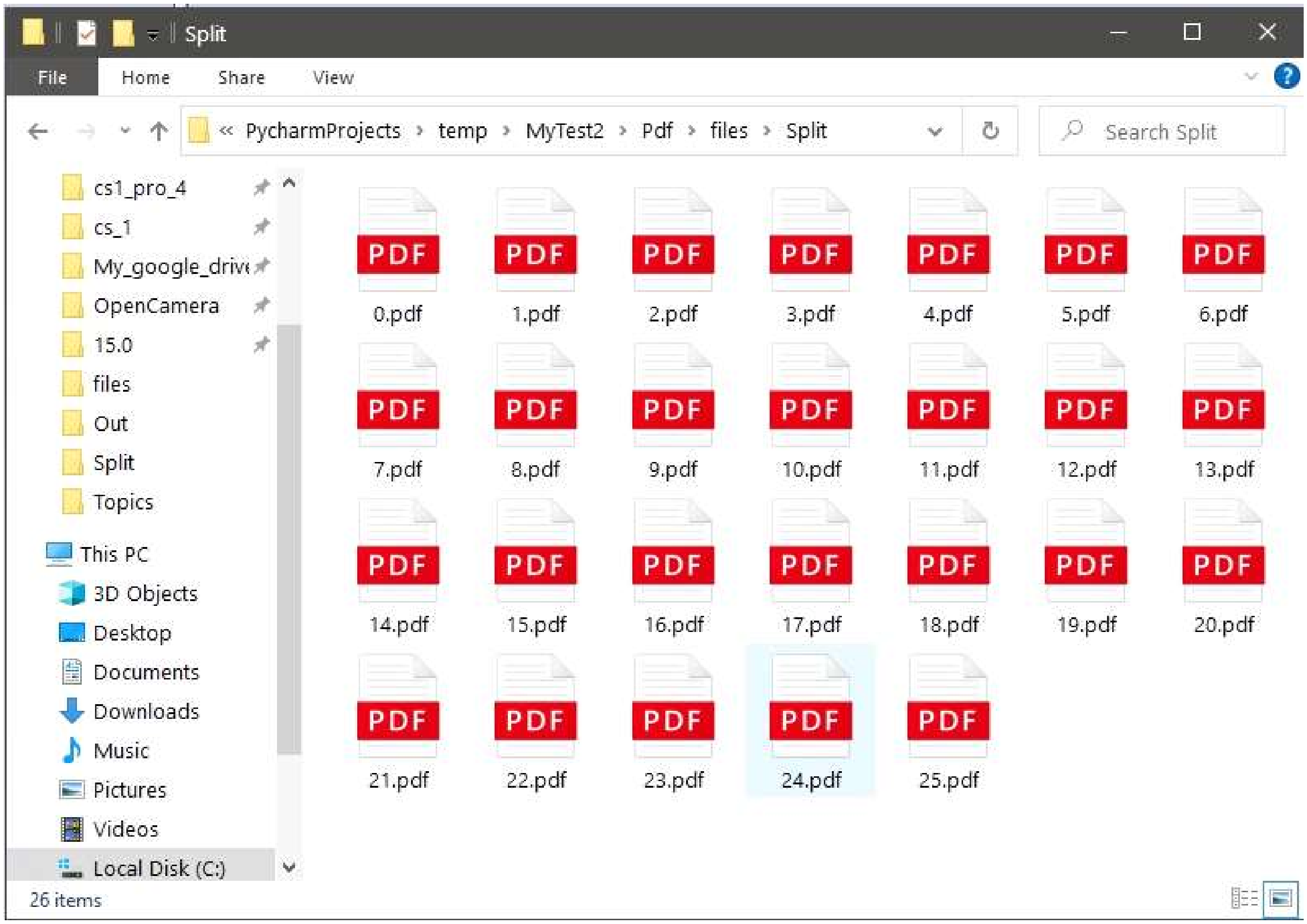
```
import PIL.Image as MyImg
#--------open image --------
```

7

```
#----------------------------------------------------------

  import PyPDF3  as MyPdf
  #---------open pdf file & create pdf reader object ------
  OrgPdfFile =open("files/test.pdf","rb")
  pdf_reader=MyPdf.PdfFileReader( OrgPdfFile , strict=False )
   # ---- create a pdf writer for new pdf file ---
  PdfWriter = MyPdf.PdfFileWriter()
  # ----- rotate all pages one by one -----
  for page in range(pdf_reader.numPages):
      pageObj = pdf_reader.getPage(page)

      pageObj.rotateClockwise(90) # rotation degree

     #------ add  rotated page object to pdf writer------

      PdfWriter.addPage( pageObj )


  #-------- create new pdf file object --------

  NewPdfFile = open( "files/rotated_Pdf_File.pdf", "wb")

  # ----- write rotated pages to new pdf file-----

  PdfWriter.write( NewPdfFile )

  # ----close the original pdf file ---------

  OrgPdfFile.close()

  # ------close the new pdf file -----------

  NewPdfFile.close()


  ==========================================
  #----------------------------------
  import PyPDF3 as MyPdf
  # ----- create pdf file merger object ----------
  PdfMerger = MyPdf.PdfFileMerger(strict=False)
  # ------ append pdf files one by one ----------
  PdfMerger.append("files/Test_Pillow.pdf")
  PdfMerger.append("files/test.pdf")
  # ----- write the combined pdf to output pdf file ------
  MergedPdfFile=open("files/MergedPdf.pdf", "wb")
  PdfMerger.write(MergedPdfFile)
  #----------------------------------

  PdfMerger.close()

  MergedPdfFile.close()
```

==

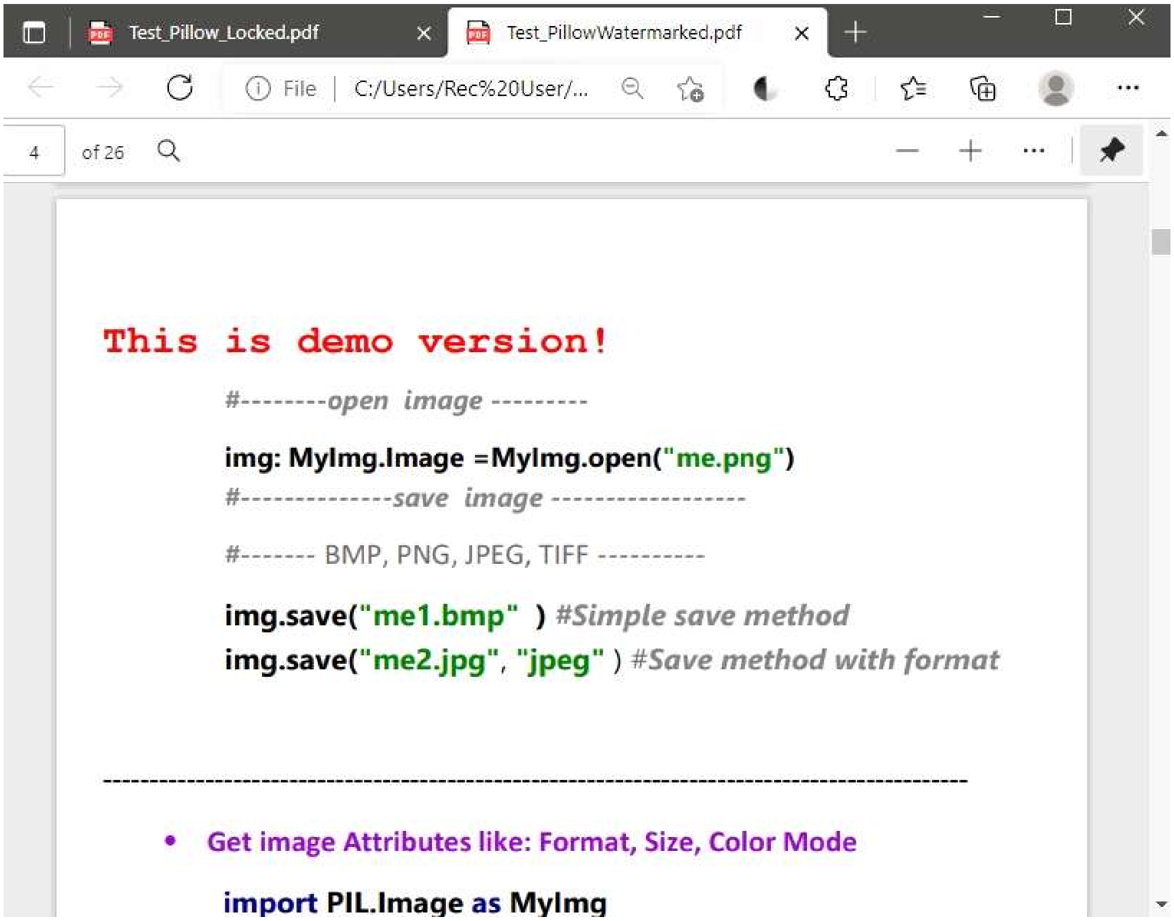
This is demo version!



```
import PyPDF3  as MyPdf

#--------open pdf file & create pdf reader object ------
OrgPdfFile =open("files/Test_Pillow.pdf","rb")
```

```
#-------split all pages one by one --------
for page in range(pdf_reader.getNumPages()):
    # --------create pdf reader object  -------
    pdf_reader = MyPdf.PdfFileReader(OrgPdfFile, strict=False)

    # ---- create a pdf writer for new pdf file ---
    PdfWriter = MyPdf.PdfFileWriter()
    # --------------
    PdfWriter.addPage(pdf_reader.getPage(page))

    # ----------create splitted file name ------------------------
    SplittedPdfFileName = str(page)+".pdf"

    # -------- create new pdf file object --------
    SplitPdfFile = open("files/split/"+SplittedPdfFileName, "wb")
    PdfWriter.write(SplitPdfFile)
    SplitPdfFile.close()

========================================
```

5. Adding watermark to PDF pages



```
#--------open  image ---------
img: MyImg.Image =MyImg.open("me.png")
#-------------save  image -----------------
#------- BMP, PNG, JPEG, TIFF ----------
img.save("me1.bmp"  ) #Simple save method
img.save("me2.jpg", "jpeg" ) #Save method with format
```

- Get image Attributes like: Format, Size, Color Mode

```
import PIL.Image as MyImg
```

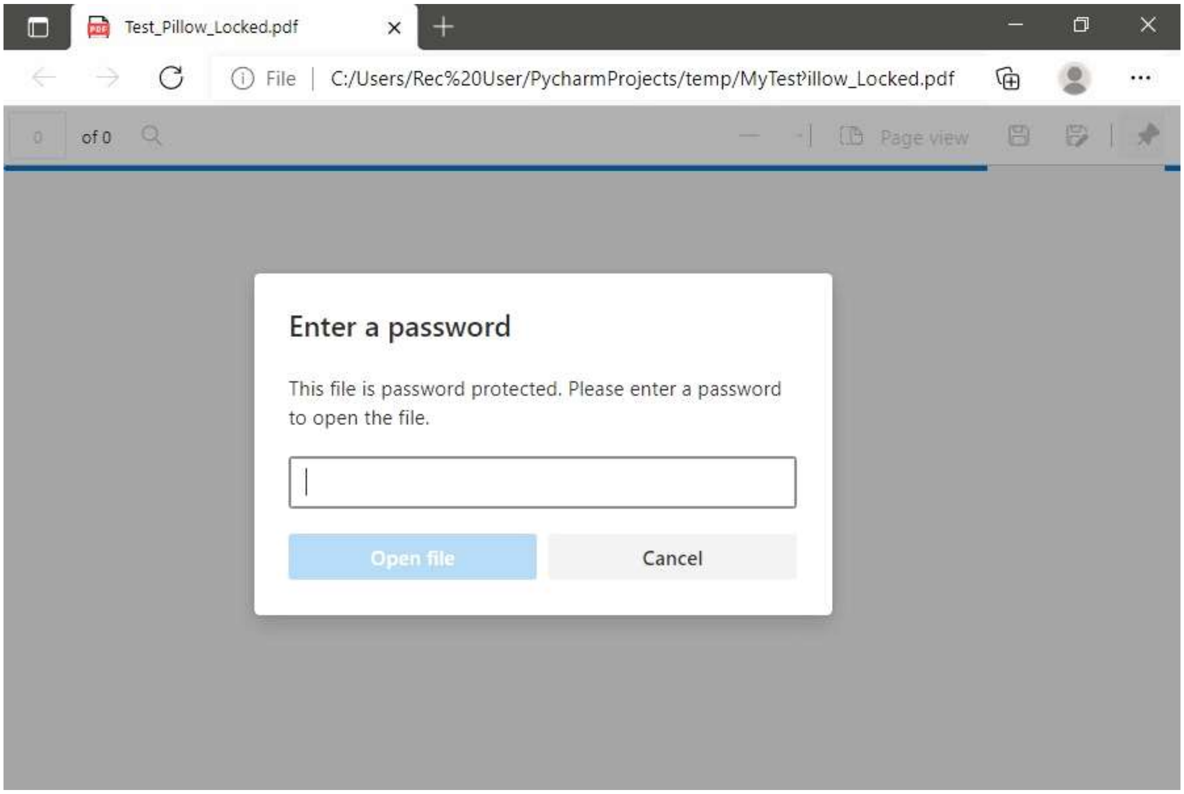
```
import PyPDF3 as MyPdf

#----------Open watermark pdf file -----------------
watermark =  "files/test.pdf"
WmPdfFile =open(watermark,"rb")
#----------Create pdf reader file for watermark------
WmFileReader = MyPdf.PdfFileReader(WmPdfFile,strict=False)
```

```
#---------open original pdf file & create reader
originalfile = "files/Test_Pillow.pdf"
OrgPdfFile =open(originalfile,"rb")
OrgPdfReader = MyPdf.PdfFileReader(OrgPdfFile,strict=False)
#---------create pdf writer for out file---------
PdfOutWrite= MyPdf.PdfFileWriter()
#---------------------------
for page in range(OrgPdfReader.getNumPages()):
   #------ get the current page of original pdf ------
   OrgPdfPage:MyPdf.pdf.PageObject=OrgPdfReader.getPage(page)
   #-------merger vm to current page of original pdf ---
   OrgPdfPage.mergePage(WaterMarkPage)
   #--------
   PdfOutWrite.addPage(OrgPdfPage)

#----------save watermarked file ------------
watermarkedfile =  "files/Test_PillowWatermarked.pdf"
OutPdfFile=open(watermarkedfile, 'wb')
PdfOutWrite.write(OutPdfFile)

#---------close all open files ----
WmPdfFile.close()
OrgPdfFile.close()
```

This is demo version!

==

6. How to Encrypt a PDF


Test_Pillow_Locked.pdf
File
C:/Users/Rec%20User/PycharmProjects/temp/MyTestPillow_Locked.pdf
0
of 0
Page view
Open file
Cancel

Enter a password

This file is password protected. Please enter a password to open the file.

```
#--------------------

import PyPDF3 as MyPdf

#--------open pdf file & create pdf reader object ------
```

```
pdf_reader = MyPdf.PdfFileReader(OrgPdfFile, strict=False)
 # ---- create a pdf writer for new pdf file ---
PdfWriter = MyPdf.PdfFileWriter()

#----- Copy source Pdf to Pdf writer -------
for page in range(pdf_reader.getNumPages()):
    PdfWriter.addPage(pdf_reader.getPage(page))

#------------Encrypt Pdf file-------------
# owner_pwd => No restrictions
# user_pwd => Custom restrictions
PdfWriter.encrypt(user_pwd="111", owner_pwd=None,
use_128bit=True)

#-----------Save Encrypted file --------------
output_pdf_file= open("files/Test_Pillow_Locked.pdf", 'wb')
PdfWriter.write(output_pdf_file)

OrgPdfFile.close()
```