Python les-materialen

# Encrypt en decrypt het pdf-bestand

pip install PyPDF2

Requirement already satisfied: PyPDF2 in c:\users\administrator\appdata\local\programs\python\python310\lib\site-packages (1.26.0)  
Note: you may need to restart the kernel to use updated packages.

## Encrypt

from PyPDF2 import PdfFileWriter, PdfFileReader

1. read the pdf file and counts the total number of pages.

file = PdfFileReader("raw\_document.pdf") # You may specify the Name of pdf here.  
total\_page = file.numPages # Count the number of pages in the pdf file  
  
out = PdfFileWriter() # Write the file to a new file using object "out"  
for page\_i in range(total\_page): # go through every page of pdf and add to new file.  
 page\_data = file.getPage(page\_i) # The data of the page  
 out.addPage(page\_data) # Add it to the new file  
  
# Write your password to encrypt the pdf  
password = input("Please enter password to encrypt the pdf: ")  
  
out.encrypt(password) # It will encrypt the pdf with given password.  
  
# Encrypted pdf will be saved with file name "encrypted\_document.pdf"  
with open("encrypted\_document.pdf", "wb") as f:  
 out.write(f) # Ceation of encrypted file.  
print("Encryption finished- file is saved to the drive!")

Encryption finished- file is saved to the drive!

## Decrypt het pdf-bestand

U hebt exact dezelfde pdf nodig die wordt gebruikt voor de codering van het bovenstaande bestand

from PyPDF2 import PdfFileWriter, PdfFileReader  
   
out = PdfFileWriter()   
file = PdfFileReader("encrypted\_document.pdf") # Read the encrypted file.  
password = input("Please enter password of the encrypted pdf: ")  
  
if file.isEncrypted: # Check the pdf whether encrypted or not.   
 file.decrypt(password) # If encrypted, decrypt it with the same password  
   
 for page\_i in range(file.numPages): # Create new file after decrypt  
 page\_data = file.getPage(page\_i) # The data of the page  
 out.addPage(page\_data) # Add it to the new file  
   
 with open("decrypted\_document.pdf", "wb") as f: # Open the "Decrypted\_file.pdf"  
 out.write(f) # Ceation of Decrypted file.  
   
 # Print success message when Done  
 print("Decryption finished- file is saved to the drive!")  
   
else:  
 print("Either file was not encrypted or was decrypted.") # If file is decrypted or not encrypted

Failed to start the Kernel.   
  
  
AttributeError: 'SelectIOLoop' object has no attribute 'asyncio\_loop'.   
  
  
View Jupyter <a href='command:jupyter.viewOutput'>log</a> for further details.