# OCR PDF Text Extractor (Python Utility)

### Overview

This project is a lightweight **Python-based OCR automation utility** that extracts readable text from **PDF files containing images or scanned screenshots**.  
It combines open-source libraries such as **pdf2image** and **pytesseract (Tesseract OCR)** to convert pages to images, perform optical character recognition (OCR), and save the extracted text to a .txt file.

While inspired by open-source and AI-generated examples, the implementation has been **adapted, tested, and refined** for practical document-processing use cases.

### Features

* Converts image-based PDF pages into images using **Poppler** via pdf2image
* Extracts text from each page using **Tesseract OCR**
* Saves all extracted text into a single output .txt file
* Supports batch processing of multi-page PDFs
* Minimal dependencies and easy to run on any platform

### Dependencies

Install required Python libraries using:

pip install pdf2image pytesseract pillow

You’ll also need to install:

* **Poppler** (for pdf2image to work)
* **Tesseract OCR Engine**

On Ubuntu/Debian:

sudo apt install poppler-utils tesseract-ocr

On Windows:

* Download Poppler → <https://github.com/oschwartz10612/poppler-windows>
* Download Tesseract → <https://github.com/UB-Mannheim/tesseract/wiki>

Make sure their installation paths are added to your system environment variables.

### How to Use

1. Place your PDF file in the same directory as the script.
2. Run:

python ocr\_pdf\_txt\_save.py input.pdf

1. The extracted text will be saved as:

input\_extracted.txt

**How It Works**

1. **PDF to Image Conversion:**  
   Converts each page of the PDF into a high-resolution image using pdf2image.
2. **OCR Processing:**  
   Uses pytesseract (Google Tesseract engine) to recognize text content in each image.
3. **Text Saving:**  
   Appends extracted text to an output file for easy search, indexing, or analysis.

**Use Cases**

* Extracting text from **scanned reports or screenshots in PDFs**
* Digitizing **archived or printed documents**
* Preparing datasets for **text mining, NLP, or data analytics**
* Automating **data entry or document processing workflows**

**Author**

**Amurdass Brojmohan**  
Python Developer | Data & Integration Consultant  
Location: Remote | Open to global contract engagements  
email: amarbrojmohan@yahoo.com

LinkedIn: www.linkedin.com/in/amar-brojmohan-pmp-8577b616

**🏷️ License**

This project is shared for educational and professional demonstration purposes.  
You are free to use, adapt, and improve it under standard open-source attribution practices.