

Rename-my-Invoices (v0.2)

A Python utility that processes invoices and receipts (PDF/images), extracts key information using OCR and LLM, and organizes them into categorized folders with standardized names.

1. Install

Create a fresh environment:

```
cd rename_my_invoices
python -m venv venv
source venv/bin/activate # On Windows: venv\Scripts\activate
pip install -r requirements.txt
```

System dependencies:

- macOS: `brew install poppler`
- Ubuntu: `sudo apt-get install poppler-utils`

2. Configure

Set up your OpenAI API key:

```
cp .env.example .env
# Edit .env to add your API key
```

You can also point the `OPENAI_API_BASE` / `OPENAI_API_KEY` variables to your compatible proxy.

3. Directory Structure

The script uses the following directory structure in your Downloads folder under `booker/`:

Required (you must create this):

- `~/Downloads/booker/raw/` - Place your unprocessed invoices/receipts here

Auto-created by the program:

- `~/Downloads/booker/creditcard/` - Processed files paid with credit cards (Visa, Mastercard, etc.)
- `~/Downloads/booker/giro/` - Processed files paid with other methods (PayPal, SEPA, etc.)
- `~/Downloads/booker/processed/` - Original files after processing (including duplicates)

Initial Setup

1. Create the raw directory:

```
mkdir -p ~/Downloads/booker/raw
```

2. Place your unprocessed invoices/receipts in the raw directory:

```
# Example: Copy some PDFs to process
cp /path/to/your/invoices/*.pdf ~/Downloads/booker/raw/
```

The program will automatically create the other directories when run.

4. File Naming Convention

Processed files are renamed using this format:

```
YY.MM.DD_method_amount_purpose.ext
```

Examples:

- 25.04.14_visa_9174_20.00USD_twilio_api.pdf
- 25.04.08_sepa_513.00EUR_finn_car.pdf
- 25.04.13_paypal_2.99EUR_apple_icloud.png

Components:

- Date: YY.MM.DD format
- Payment method: visa, mastercard, paypal, sepa, etc.
 - For credit cards: includes last 4 digits if available
 - For PayPal: includes email if available
- Amount: includes currency (EUR, USD, etc.)
- Purpose: company_purpose format (max 8 chars for purpose)

5. Usage

Try a dry run first:

```
python rename_my_invoices.py --dry-run
```

If the output looks good, process the files:

```
python rename_my_invoices.py
```

Options

Flag	Effect
<code>--dry-run</code>	Preview the proposed changes without modifying anything
<code>--log rename_log.jsonl</code>	Append detailed JSON logs for each processed file

6. How it works

1. Text Extraction

- PDF text → `pdfminer.six`
- If no text found → Convert to image with `pdf2image`
- Image OCR → `easyocr` (supports English and German)

2. LLM Processing

- Sends extracted text to GPT-4o
- Extracts date, payment method, amount, and purpose
- Normalizes fields for consistent naming

3. File Organization

- Sorts into `creditcard/` or `giro/` based on payment method
- Moves original to `processed/`
- Handles duplicates by adding `"duplicate_"` prefix

Payment Method Sorting

Files are sorted based on the payment method:

creditcard/ folder:

- Visa payments
- Mastercard payments
- Generic card payments
- Unknown payment methods

giro/ folder:

- PayPal payments
- SEPA transfers
- All other payment methods

7. Duplicate Handling

When a file would create a duplicate in the target directory:

- Original stays in place
- New file is moved to `processed/` with `"duplicate_"` prefix

- Both actions are logged

Happy organizing!