



# HR FILE AUTOMATION TOOL – USER GUIDE

Documentation/Manual/Guideline



JULY 13, 2025

ABDUR RAHMAN KHAN

 <https://www.linkedin.com/in/seark>

## Overview

---

This user guide provides detailed instructions on how to use the HR File Automation Tool, which simplifies repetitive file management tasks such as renaming and format conversion. It is designed for HR teams, administrative assistants, and IT support professionals who need offline automation.

### Directory Structure

HR\_File\_Automation\_Tool/

- └─ scripts/ - Python automation scripts
- └─ input/
  - └─ sample\_files/ - Place files here for processing
  - └─ errors/ - Files with naming or format errors
- └─ output/
  - └─ renamed/ - Renamed files saved here
  - └─ converted/ - Format-converted files saved here
- └─ logs/ - Renaming and conversion logs
- └─ documentation/
  - └─ user\_guide.docx - This guide
  - └─ screenshots/ - Before/after snapshots
- └─ requirements.txt - Install dependencies
- └─ README.md - Project overview
- └─ LICENSE.txt - Project usage rights

## **Tools Needed (Offline)**

- Python 3.8 or higher (Windows 10 Pro compatible)
- No internet required after initial setup
- All libraries listed in requirements.txt
- Can be bundled into a virtual environment or run as-is

## **Features**

### **1 Bulk File Renamer**

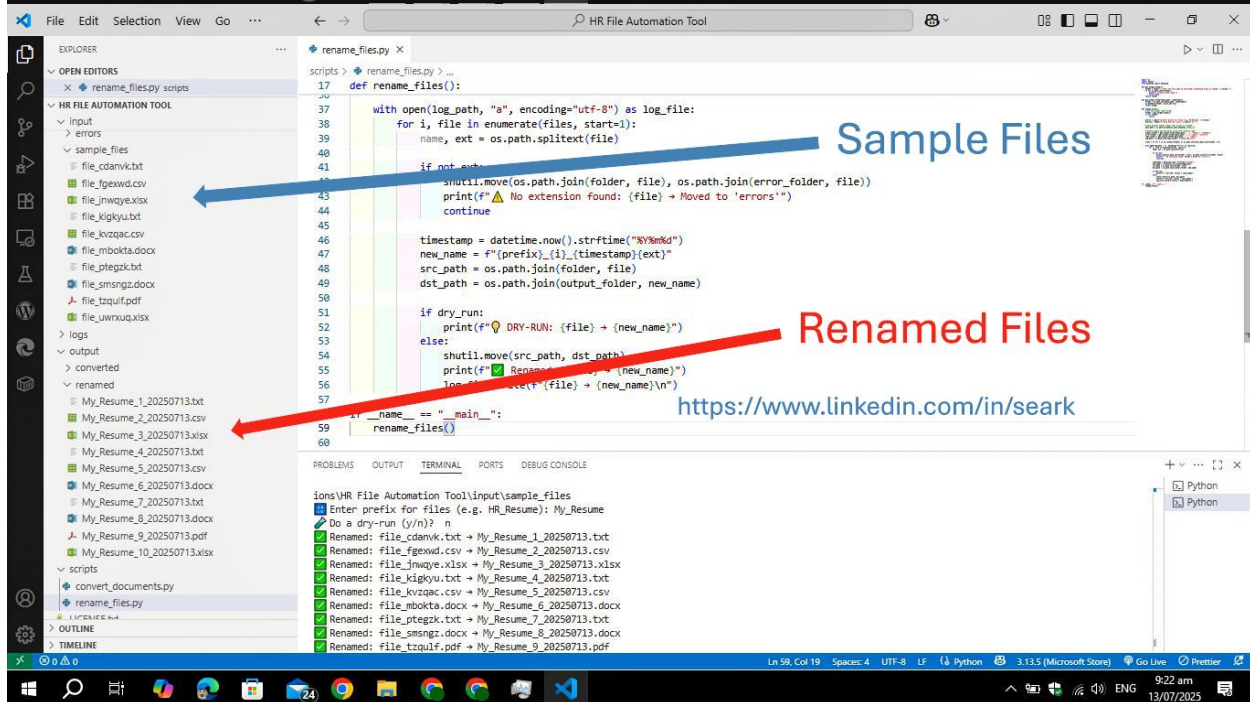
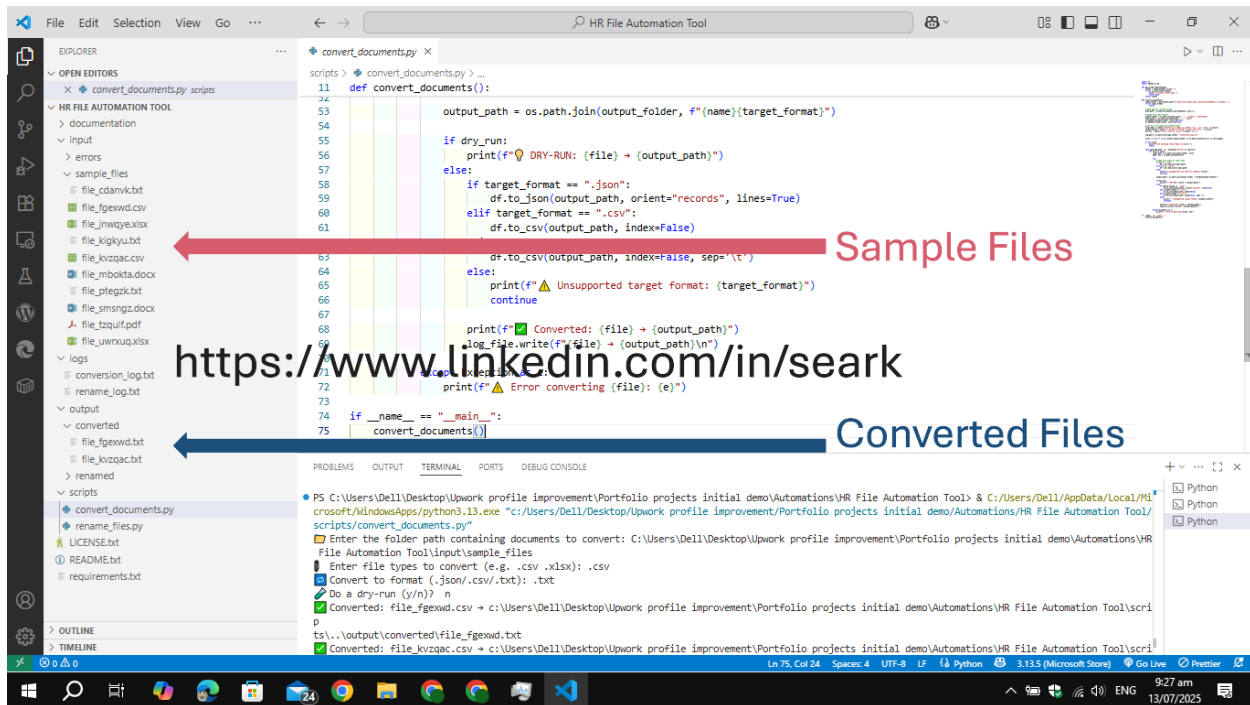
- Prompts user for folder path
- Renames files using HR-friendly naming patterns
- Moves invalid files to a separate 'errors' folder
- Offers dry-run preview mode for safe testing
- Logs all renaming actions to logs/rename\_log.txt

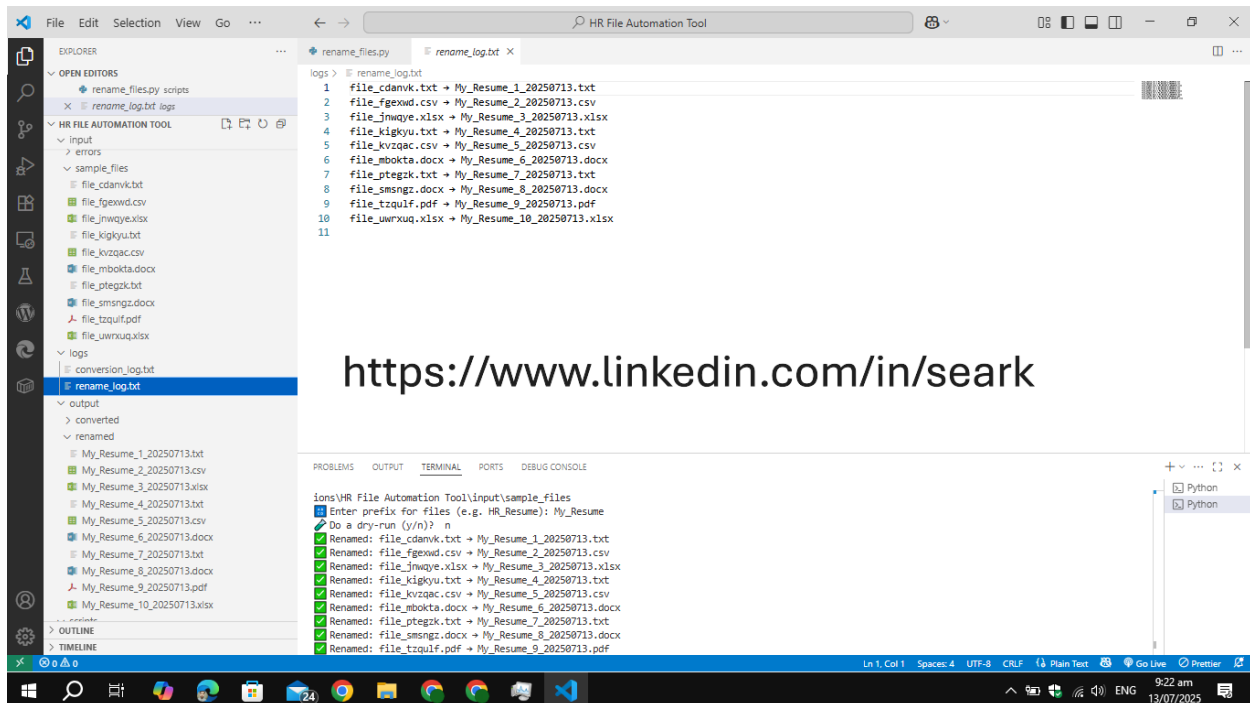
### **2 Document Converter**

- Converts CSV, XLSX, DOCX, PDF, TXT into standardized formats like TXT or JSON
- Batch processes entire folders
- User selects format and destination
- Invalid files are moved to 'input/errors'
- Logs conversion actions to logs/conversion\_log.txt

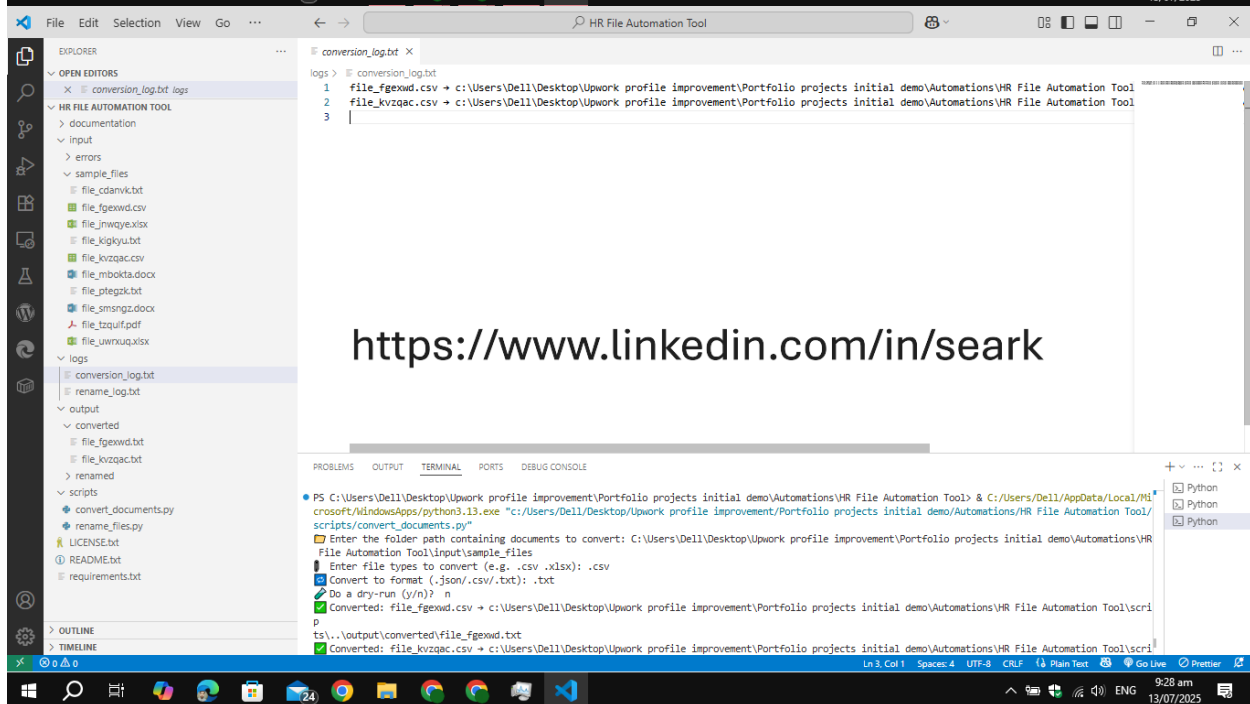
## **How to Use This Tool**

1. Download and extract the ZIP file from GitHub.
2. Install Python 3.8+ if not already installed.
3. Open a terminal or command prompt inside the project folder.
4. Run: `pip install -r requirements.txt`
5. To use the renamer, run: `python scripts/rename_files.py`
6. To use the converter, run: `python scripts/convert_documents.py`
7. Optionally set up a virtual environment for clean portability.
8. If you prefer an IDE like VS Code, ensure dependencies are installed and run directly from the editor.





<https://www.linkedin.com/in/seark>



<https://www.linkedin.com/in/seark>