

Sequential File Renamer

A cross-platform utility for batch renaming files with sequential numbering and consistent file extensions.

Overview

Sequential File Renamer allows you to quickly rename multiple files with a consistent, sequential naming pattern. It's ideal for organizing photos, documents, or any collection of files that would benefit from a uniform naming scheme.

Features

- **Sequential Renaming with Adaptive Padding**
 - Single digit padding for fewer than 10 files (e.g., photo_1.jpg)
 - Double digit padding for 10-99 files (e.g., photo_01.jpg)
 - Triple digit padding for 100+ files (e.g., photo_001.jpg)
- **File Selection Flexibility**
 - Process entire directories at once
 - Select specific files to rename
 - Work with files across different folders
- **Extension Handling**
 - All extensions automatically converted to lowercase
 - Normalization of non-standard extensions to common formats
 - Example: .JPEG → .jpg, .Tiff → .tif, .Markdown → .md
- **Additional Options**
 - Add date prefix in YYYYMMDD format
 - Filter files by extension

Installation

Prerequisites

- Python 3.6+
- Tkinter (usually included with Python)
- Pydantic (installed automatically by installer)

Linux Installation

```
# Extract the package, then:  
cd file-renamer  
chmod +x install.sh  
./install.sh
```

For system-wide installation:

```
sudo ./install.sh
```

Windows Installation

Simply download and run the Windows executable (.exe file).

Manual Installation

```
# Clone the repository
git clone https://github.com/yourusername/file-renamer.git
cd file-renamer

# Install dependencies
pip install pydantic

# Run the application
python3 file_renamer.py
```

How to Use

Step 1: Select Files

Choose one of two selection methods:

- **Directory Mode:**
 1. Click “Browse Directory...”
 2. Select a folder containing files to rename
 3. Choose “Process entire directory”
- **File Selection Mode:**
 1. Click “Select Files...”
 2. Choose individual files (use Ctrl/Shift for multiple selections)
 3. Choose “Process selected files only”

Step 2: Configure Options

1. **Enter Base Name**
 - Type the text you want to use as the base name (e.g., “photo”)
 - Result: “photo_1.jpg”, “photo_2.jpg”, etc.
2. **Select Optional Settings**
 - Include Date: Adds current date (YYYYMMDD_) as prefix
 - Normalize Extensions: Converts uncommon extensions to standard ones
 - Filter by Extension: Specify which file types to process (e.g., “jpg,png,txt”)

Step 3: Preview and Apply

1. Click “Generate Preview” to see how files will be renamed
2. Review the changes in the preview panel
3. Click “Rename Files” to apply the changes

Extension Normalization

When enabled, these file extensions are automatically converted:

Original	Normalized	Original	Normalized
.jpeg	.jpg	.mpeg	.mpg
.tiff	.tif	.mov	.mp4
.htm	.html	.text	.txt
.markdown	.md	.midi	.mid

Examples

Basic Sequential Renaming:

Before:		After (base name "photo"):
IMG_1234.jpg	→	photo_1.jpg
20050123.jpg	→	photo_2.jpg
scan0003.jpg	→	photo_3.jpg

With Date Prefix:

Before:		After (base name "vacation"):
DSC_9876.jpg	→	20240311_vacation_1.jpg
IMG_2468.jpg	→	20240311_vacation_2.jpg

With Extension Normalization and Case Conversion:

Before:		After (base name "image"):
picture.JPEG	→	image_1.jpg
photo.Tiff	→	image_2.tif
file.TXT	→	image_3.txt

Troubleshooting

Installation Issues

- **Command not found:** Ensure your PATH includes the bin directory
- **Missing dependencies:** Run `pip install pydantic` manually
- **Permission errors:** Use `sudo` for system-wide installation

Runtime Issues

- **File access errors:** Check folder/file permissions
- **No files found:** Verify directory path or file selection
- **Cannot rename:** Ensure files aren't open in other applications

Uninstallation

To remove the application:

```
# For user installation:  
file-renamer-uninstall
```

```
# For system-wide installation:  
sudo file-renamer-uninstall
```

License

This project is licensed under the MIT License - see the LICENSE file for details.

Contributing

Contributions are welcome! Please feel free to submit a Pull Request.