File Renamer

A cross-platform utility to easily rename multiple files using patterns, regular expressions, and more.

Features

- **Multiple Renaming Methods**:
  - Add prefix to filenames
  - Add suffix to filenames
  - Replace text in filenames
  - Use regular expressions for advanced renaming
- **Additional Options**:
  - Include date stamps (YYYYMMDD format)
  - Add sequential numbering
  - Filter files by extension
- **User-Friendly GUI**:
  - Preview changes before applying
  - Easy directory selection
  - Status updates during operations

Installation

Linux

1. Download the installation package

2. Extract the package contents

3. Navigate to the file-renamer directory:

   ```
   cd file-renamer
   ```

4. Run the installer script:

   ```
   chmod +x install.sh
   ./install.sh
   ```

For system-wide installation (requires administrator privileges):

```
chmod +x install.sh
sudo ./install.sh
```

What the Installer Does The installer performs these steps: 1. Checks for and installs prerequisites (Python, pip, Tkinter) 2. Creates the installation directory structure 3. Copies the application files 4. Creates a launcher script in the appropriate bin directory 5. Creates a desktop entry for application menus 6. Installs an uninstaller script and creates a command to run it 7. Optionally builds a standalone executable with PyInstaller

Uninstalling   To uninstall:

```
file-renamer-uninstall
```

Or for system-wide installations:

```
sudo file-renamer-uninstall
```

You can also uninstall directly from the source directory:

```
cd file-renamer
chmod +x uninstall.sh
./uninstall.sh
```

Windows

1. Download the Windows executable (.exe file)
2. Run the executable to start the application

Manual Installation

Prerequisites

- Python 3.6 or higher
- Tkinter (usually included with Python)

Installation Steps

1. Clone this repository:

   ```
   git clone https://github.com/yourusername/file-renamer.git
   ```

2. Navigate to the directory:

   ```
   cd file-renamer
   ```

3. Run the application:

   ```
   python3 file_renamer.py
   ```

Directory Structure

```
file-renamer/

    file_renamer.py              # Main application script
    icon.png                     # Application icon (PNG format)
    icon.ico                     # Application icon (Windows format)
    LICENSE                      # License file
    README.md                    # Documentation
    install.sh                   # Installation script
    uninstall.sh                 # Uninstallation script (optional, also created during install
```

The install.sh and uninstall.sh scripts are designed to be run from within the file-renamer directory.

Usage Guide

1. **Select a directory**: Click "Browse…" to choose a folder containing the files you want to rename.

2. **Choose a renaming pattern**:

   - Add Prefix: Adds text to the beginning of filenames
   - Add Suffix: Adds text before the extension
   - Replace Text: Substitutes specific text with new text
   - Regular Expression: Uses regex patterns for advanced renaming

3. **Set additional options**:

   - Include Date: Adds the current date in YYYYMMDD format
   - Include Numbering: Adds sequential numbers to files
   - Filter by Extension: Enter comma-separated extensions (e.g., jpg,png,txt)

4. **Preview changes**: Click "Generate Preview" to see how files will be renamed

5. **Apply changes**: When satisfied, click "Rename Files" to apply the changes

Building from Source

Building for Linux

```
# Install PyInstaller
pip3 install pyinstaller

# Navigate to the source directory
cd file-renamer

# Build the executable
pyinstaller --name FileRenamer --windowed --onefile file_renamer.py
```

Building for Windows

```
# Install PyInstaller
pip install pyinstaller

# Navigate to the source directory
cd file-renamer
```

```
# Build the executable
pyinstaller --name FileRenamer --windowed --onefile --icon=icon.ico file_renamer.py
```

Advanced Packaging (Linux)

Creating DEB Package (Debian/Ubuntu)

1. Install necessary tools:

   ```
   sudo apt-get install build-essential debhelper dh-python
   ```

2. Create a Debian packaging structure (basic steps):

   ```
   mkdir -p debian/DEBIAN
   cat > debian/DEBIAN/control << EOF
   Package: file-renamer
   Version: 1.0
   Section: utils
   Priority: optional
   Architecture: all
   Depends: python3, python3-tk
   Maintainer: Your Name <your.email@example.com>
   Description: File Renaming Utility
    A cross-platform utility to easily rename multiple files.
   EOF
   ```

Creating RPM Package (Fedora/RHEL)

1. Install necessary tools:

   ```
   sudo dnf install rpm-build rpmdevtools
   ```

2. Set up RPM build environment:

   ```
   rpmdev-setuptree
   ```

3. Create an RPM spec file (basic example):

   ```
   cat > ~/rpmbuild/SPECS/file-renamer.spec << EOF
   Name:           file-renamer
   Version:        1.0
   Release:        1%{?dist}
   Summary:        File Renaming Utility

   License:        MIT
   URL:            https://github.com/yourusername/file-renamer
   Source0:        %{name}-%{version}.tar.gz

   BuildArch:      noarch
   Requires:       python3, python3-tkinter
   ```

```
%description
A cross-platform utility to easily rename multiple files using patterns,
regular expressions, and more.

%prep
%setup -q

%install
mkdir -p %{buildroot}/opt/file-renamer
cp -a * %{buildroot}/opt/file-renamer/

mkdir -p %{buildroot}%{_bindir}
cat > %{buildroot}%{_bindir}/file-renamer << EOF2
#!/bin/bash
cd /opt/file-renamer && python3 file_renamer.py "\$@"
EOF2
chmod +x %{buildroot}%{_bindir}/file-renamer

mkdir -p %{buildroot}%{_datadir}/applications
cat > %{buildroot}%{_datadir}/applications/file-renamer.desktop << EOF2
[Desktop Entry]
Name=File Renamer
Comment=Rename files easily
Exec=/usr/bin/file-renamer
Icon=/opt/file-renamer/icon.png
Terminal=false
Type=Application
Categories=Utility;FileTools;
EOF2

%files
%attr(755, root, root) %{_bindir}/file-renamer
%{_datadir}/applications/file-renamer.desktop
/opt/file-renamer

%changelog
* Wed Mar 11 2025 Your Name <your.email@example.com> - 1.0-1
- Initial package
EOF
```

Troubleshooting

Common Installation Issues

- **Command not found**: If you get "command not found" errors, make sure your PATH includes the bin directory
- **Missing Python/Tkinter**: Ensure Python 3 and Tkinter are installed on your system
- **Permission errors**: Check if you need to run with sudo for system-wide installation
- **Application doesn't appear in menu**: Try logging out and back in to refresh the application menu

Runtime Issues

- **File access errors**: Ensure the application has read/write permissions for the selected directory
- **UI glitches**: Some themes may not display correctly; try changing your system theme
- **Regular expression errors**: Double-check your regex patterns for syntax errors

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.