

Snippet Manager Script

This `snippet-manager.sh` script provides a convenient way to manage and quickly copy frequently used text snippets or file contents to your clipboard using `fzf` for fuzzy searching. It's designed to streamline your workflow by allowing you to search through a collection of snippets by name or tags and then copy the relevant content.

Features

- **Fuzzy Search:** Quickly find snippets using `fzf`.
- **Clipboard Integration:** Copies selected snippet content or file content directly to your system clipboard.
- **File Content Support:** If a snippet's content is a valid file path, the script will copy the file's content instead of the path itself.
- **Newline Transformation:** Automatically converts `*` characters within a snippet to newlines when copying, useful for multi-line snippets.
- **Responsive Preview:** Displays a preview of the snippet or file content in `fzf` before selection.

Prerequisites

Before using this script, ensure you have the following installed:

- `fzf`: A command-line fuzzy finder.
 - Installation: `git clone --depth 1 https://github.com/junegunn/fzf.git ~/.fzf && ~/.fzf/install` (or use your system's package manager).
- `awk`: A text processing tool (usually pre-installed on Linux/macOS).
- `sed`: A stream editor (usually pre-installed on Linux/macOS).
- `tr`: A utility for translating or deleting characters (usually pre-installed on Linux/macOS).
- **Clipboard Utility:**
 - Wayland: `wl-copy` (from `wl-clipboard` package).
 - X11: `xclip`.

Setup

1. **Save the Script:** Save the provided `snippet-manager.sh` script to a convenient location (e.g., `~/bin/snippet-manager.sh`).

2. Make Executable:

```
chmod +x ~/bin/snippet-manager.sh
```

3. **Create snippet file:** The script expects a CSV file named `snippets.csv` at `~/.local/share/snippets/snippets.csv`.

Create the directory:

```
mkdir -p ~/.local/share/snippets
```

Create the `snippets.csv` file:

```
touch ~/.local/share/snippets/snippets.csv
```

4. **Configure snippets.csv:** The `snippets.csv` file uses `###` as a delimiter and should follow this format:

Snippet Name###tags###snippet content or file path

- *Snippet Name:* A descriptive name for your snippet.
- *tags:* Keywords to help you search (e.g., `bash`, `utility`, `git`).
- *snippet content or file path:*
 - The actual text you want to copy.
 - **OR** a valid file path (e.g., `/home/user/my_long_script.sh`). If it's a file path, the script will copy the *content* of that file.

Example snippets.csv entries:

```
Hello World###greeting,test###echo "Hello, World!"
My Long Script###script,bash,utility###/home/user/scripts/long_script.
Multi-line Example###text,example###Line 1*Line 2*Line 3
Empty File Example###test,empty###/home/user/empty_file.txt
```

Usage

You can run the script with different commands to perform specific actions:

- **No Command** (Default Behavior):

```
snippet-manager.sh
```

When run without any arguments, the script will launch `fzf`. This allows you to interactively search, preview, and select a snippet from your `snippets.csv` file to copy its content (or the content of a referenced file) to your clipboard.

- **add Command:**

```
snippet-manager.sh add
```

This command allows you to interactively add a new snippet to your `snippets.csv` file. You will be prompted for the snippet name, tags, and the content (or file path). For multi-line content, you can type your snippet and press `Ctrl+D` on a new line to finish input.

- **edit Command:**

```
snippet-manager.sh edit
```

This command opens your `snippets.csv` file in your preferred text editor (determined by the `$EDITOR` environment variable, or `nano` if `$EDITOR` is not set). This provides a quick way to manually add, modify, or delete snippets directly in the CSV file. Remember to save and close the editor after making changes.

- **help Command:**

```
snippet-manager.sh help
```

This command displays a concise help message, outlining the script's usage, available commands, and the expected format of the `snippets.csv` file.

- **tags Command:**

```
snippet-manager.sh tags
snippet-manager.sh tags > snippet-manager-tags.txt
snippet-manager.sh tags | bat
```

This command extracts and lists all unique tags found in your `snippets.csv` file. The tags are sorted alphabetically. You can pipe or redirect the result as you wish.

- **Unknown Command:**

```
snippet-manager.sh [any_invalid_command]
```

If you provide an argument that is not a recognized command (e.g., `snippet-manager.sh foo`), the script will print an error message indicating the unknown command, then display the standard help message, and exit with a non-zero status.

Interaction

- **Search:** Type to filter snippets by name or tags.
- **Navigate:** Use arrow keys (Up / Down) to move between results.
- **Preview:** The right pane will show a preview:
 - If the third field is a file path, it will display "--- Snippet in a File ---" followed by the file's content; if the file is empty, then it will display "--- Empty file ---".

- Otherwise, it will display "--- Snippet ---" followed by the first 200 characters of the snippet content.
- **Select & Copy:** Press `Enter` to copy the selected snippet's content (or file's content) to your clipboard.
- **Exit:** Press `Esc` or `Ctrl+C` to exit without copying.

How it Works (Copy Logic)

When you select a snippet and press `Enter`, the script performs the following checks on the third field (the snippet content/path):

1. Is it a valid, non-empty file?

- If **Yes**: The *entire content* of the file is copied to the clipboard.
- If **No** (but it's an existing, empty file): A message "Selected file is empty." is printed to `stderr`, and nothing is copied.

2. Is it not a file (or the file check failed)?

The script then checks if the snippet content contains a `*` character anywhere.

- If **Yes**: All `*` characters in the snippet are replaced with newline characters (`\n`), and the resulting multi-line text is copied to the clipboard.
- If **No**: The snippet content is copied to the clipboard as is.