File System Tools Cheatsheet (Linux)

A comprehensive cheat sheet covering essential Linux file system tools like `ls`, `cp`, `mv`, and `rm`, including options, examples, and best practices for efficient file management.

ls - Listing Files and Directories

Basic Usage

`ls`	Lists files and directories in the current directory.
`ls <directory>`	Lists files and directories in the specified directory.
`ls -l`	Lists files in long format, showing permissions, size, modification date, etc.
`ls -a`	Lists all files, including hidden files (those starting with `.`).
`ls -t`	Sorts files by modification time (newest first).
`ls -r`	Reverses the order of the listing.

Combining Options

`ls -la`	Lists all files (including hidden) in long format.
`ls -lt`	Lists files in long format, sorted by modification time.
`ls -ltr`	Lists files in long format, sorted by modification time (oldest first).
`ls -lh`	Lists files in long format, with file sizes in human-readable format (e.g., KB, MB, GB).
`ls -d`	List directories - useful with wildcards.
`ls -F`	Appends a character to each filename indicating the file type (* for executable, / for directory, @ for symbolic link, = for socket).

Advanced Usage

`ls -R`	- Lists subdirectories encountered recursively.
`ls -S`	- Sort by file size, largest first.
`ls -i`	- Show the index number of each file.
`ls --color[=WHEN]`	- Control whether color is used to distinguish file types. `WHEN` can be `always` (default if omitted), `auto`, or `never`.
`ls -b`	- Escape non graphic characters as octal numbers.

cp - Copying Files and Directories

Basic Copying

`cp <source> <destination>`	Copies a file or directory from source to destination.
`cp file1 file2 file3 <directory>`	Copies multiple files to a directory.
`cp -r <source_dir> <destination_dir>`	Copies a directory recursively (including all files and subdirectories).
`cp -i <source> <destination>`	Interactive mode. Asks for confirmation before overwriting existing files.
`cp -p <source> <destination>`	Preserves the original file's mode, ownership, and timestamps.

Advanced Copying

`cp -u <source> <destination>`	- Copy only when the SOURCE file is newer than the destination file or when the destination file is missing.
`cp -v <source> <destination>`	- Verbose mode. Shows files as they are copied.
`cp -s <source> <destination>`	- Make symbolic links instead of copying.
`cp -l <source> <destination>`	- Make hard links instead of copying.
`cp --backup[=CONTROL] <source> <destination>`	- Back up each existing destination file. The CONTROL argument specifies the backup suffix.

Examples

Copying a file:
`cp my_file.txt /home/user/documents/`

Copying a directory recursively:
`cp -r my_directory /home/user/backup/`

Copying multiple files with confirmation:
`cp -i file1.txt file2.txt /tmp/`

mv - Moving and Renaming Files and Directories

Basic Moving and Renaming

`mv <source> <destination>`	Moves or renames a file or directory from source to destination.
`mv file1 file2 file3 <directory>`	Moves multiple files to a directory.
`mv -i <source> <destination>`	Interactive mode. Asks for confirmation before overwriting existing files.

Moving Options

`mv -n <source> <destination>`	No clobber. Do not overwrite an existing file. Useful for scripts.
`mv -v <source> <destination>`	Verbose. Shows files as they are moved.
`mv -u <source> <destination>`	Move only when the SOURCE file is newer than the destination file or when the destination file is missing.

Examples

Renaming a file:
`mv old_file.txt new_file.txt`

Moving a file to a different directory:
`mv my_file.txt /home/user/documents/`

Moving multiple files to a directory:
`mv file1.txt file2.txt /tmp/`

Moving a directory:
`mv my_directory /opt/`

rm - Removing Files and Directories

Basic Removal

`rm <file>`	Removes a file.
`rm -i <file>`	Interactive mode. Asks for confirmation before deleting.
`rm -f <file>`	Force removal. Ignores nonexistent files and suppresses prompts.

Removing Directories

`rmdir <directory>`	Removes an empty directory.
`rm -r <directory>`	Removes a directory and its contents recursively.
`rm -rf <directory>`	Force removal of a directory and its contents recursively. Use with caution!

Examples

Removing a file:
`rm my_file.txt`

Removing a directory recursively:
`rm -r my_directory`

Force removing a directory and its contents:
`rm -rf my_directory`

Removing multiple files:
`rm file1.txt file2.txt file3.txt`

Important Notes

Caution: `rm -rf /` will delete everything on the system if run as root. Be extremely careful when using `rm` with wildcards or the `-r` and `-f` options.

Best Practice: Use `rm -i` to confirm each deletion, especially when using wildcards.