

man: shows the manual page for any given command

example: 'man man' (shows manual page for man), 'man ls' (shows manual page for ls)

ssh: connect to a remote system over an encrypted (usually OpenSSH) connection

flags: -p (specifies port)

example: ssh jay@10.143.90.2 -p 2000 (connects to remote server as user jay on port 2000)

ls: list directory contents

flags: -l (long format), -a (show hidden files), -h (human-readable sizes),

-R (recursive)

example: ls -lha . (show all files in long, human readable format in current directory)

ls -R src (recursively show all files in the src directory)

cd: change the current working directory

example: cd src (switch the working directory to the src folder)

cd ~ (switch the working directory to the home directory)

cat: displays file contents by writing to stdout

flags: -n (number of lines), -b (number non-blank lines), -E (show line ends)

example: cat -n 3 readme.txt (display the first 3 lines of the file readme.txt)

cat -b 6 example.md (display the first 6 non-blank lines of the file example.md)

file: determine file type based on content (NOT EXTENSION)

flags: -b (brief, omit filename), -i (show MIME type), -z (examine compressed file)

example: file -b main (determine the file type of main, only display type info)

du: display disk usage of files/directories

flags: -h (human-readable), -s (summary only), -a (include files), -c (grand total)

example: du -ha (display sizes of all directories and files in the current directory)

find: search for file/directory using REGEX

flags: -name (match name in subdirectories), -type (match type, f=file, d=directory),

-exec (run commands on found files)

example: find -name a.out (find all files named a.out in current and sub directories)

find *.c (find all c files in the current directory)

find . -type f -exec file {} \; (find all files in current directory, then execute the file command on every one)

mkdir: create a directory

flags: -p (create parent directories)

example: mkdir include (make a directory called include in the current directory)

mkdir src/java (make a subdirectory java in the directory src, create src if it doesn't exist)

rmdir: remove empty directories

flags: -p (remove parent directories if empty)

example: rmdir src (remove the directory src if empty)

rmdir -p main/interfaces (remove the directory interfaces if empty, then main if also empty)

touch: create empty file

example: touch main.c (create a file named main.c in the working directory)

touch src/algorithm.java (create a file named algorithm.java in the src directory)

rm: remove files or directories

flags: -r (recursive), -f (force), -v (verbose)

example: rm temp.txt (remove the file temp.txt from the working directory)

rm -rf ./bin (remove all files from the bin directory, by force)

mv: moves or rename files and directories

flags: -i (warn on overwrite), -f (force overwrite), -n (no overwrite)

example: mv text.txt contacts.txt (rename text.txt to contacts.txt)

mv main.c ../src (move main.c to the folder in the parent directory called src)

grep: search for patterns in file/directory using REGEX

flags: -i (ignore case), -r (recursive search), -n (show line number),

-v (invert match), -l (list matching files)

example: grep -r 'void main()' . (find matching pattern in current and sub-directories)

grep -n 'hello' text.txt (print lines that contain the pattern in file text.txt)

'|': pipe/send output of one command as input to another

example: ls | grep txt (filter ls output for lines containing txt)

cat contacts.txt | sort (get all lines in contacts.txt, then sort and show them)

'>': (over)write output to a file

example: cat contacts.txt | sort > sorted_contacts.txt (get all the lines from

contacts.txt, sort them, then save the output to a file called sorted_contacts.txt)

'>>': append output to a file

example: cat new_info.txt >> logs.txt (append new_info.txt to the end of logs.txt)

'<': take input from a file

example: grep 'John Smith' < contacts.txt (find matching pattern in contacts.txt)

'2>', '&>': error redirection, output stderr(2>) or both stdout/stderr(&>)

example: find /var/log -type f -name '*.log' > files.txt 2> errors.txt (find files

matching the name *.log, send output to files.txt, send errors to errors.txt)

'2>&1': merge outputs, redirect stderr to stdout

example: du -sh /home/* /nonexistent 2>&1 | grep 'No such file' (summarize sizes for home and nonexistent directories, then merge into stdout, then filter only the error messages)

tar: create and extract archive files

flags: -c (create archive), -x (extract files), -f (specify filename of archive), -z (use gzip compression), -j (use bzip2 compression), -v (display verbose information)

example: tar -cvf cfiles.tar . (create an archive with all files in the current directory)

tar -xf file.tar (extracts files from file.tar archive)

tar -cvjfsource.bz2 *.c (archive all .c files in current directory in bzip2 format)