Linux CLI Cheat-Sheet (Rahul's List Explained)

#	Command	What it does / why you care
1	pwd	Print absolute path of present working directory.
2	ls	List names of files & dirs (alphabetical).
3	ls -R	Recurse into sub-dirs and list everything.
4	ls -t	Sort by modification time, newest first.
5	ls -l	Long format: perms, owner, size, date, name.
6	ls -lt	Long listing sorted by time.
8	ls -la	Long + show hidden files (dot-files).
9	ls -lRa	Long, recursive, all files (hidden too).
10	ls -lr	Reverse whatever order is currently used.
11	ls -s	Show allocated blocks next to each name.
12	ls *.js	List only JavaScript files.
13	ls Zoo*	List files whose names start with "Zoo".
14	ls ..	List parent directory without moving.
15	cd	Change directory (home dir if no arg).
16	cd ..	Go up one level.
17	cd ../../	Go up two levels.
18	touch f	Create empty file or update its timestamp.
19	cat f	Dump whole file to terminal.
20	cat > f	Type text; Ctrl-D saves & overwrites file.
21	cat >> f	Append typed text to file.
22	mkdir d	Create directory "d".
23	mkdir d && cd d	Create and step inside in one go.
24	mkdir -p a/b/c	Create nested directory path.
25	mv old new	Rename or move file/dir.
26	mv path/old path/new	Same as 25 with full paths.
27	cp f new	Copy file.
28	cp -r d new	Copy directory recursively.
29	rm f	Delete file (irreversible).
30	rm -r d	Delete directory and everything inside.
31	chmod ugo-rwx f	Remove all permissions from everybody.
31b	chmod -R ugo-rwx d	Same, recursive for directory.
32	chmod u+x f	Give owner execute permission.
33	chmod g+wx f	Give group write & execute.
34	chmod u-x f	Remove owner execute.
35	chmod 664 f	Numeric: rw-rw-r--.
36	echo 'msg'	Print message to terminal.
37	head f	First 10 lines.
38	tail f	Last 10 lines.
39	head -20 f	First 20 lines.
40	tail -n +25 f | head -5	Lines 25-29 only.
41	wc f	Lines, words, characters count.
42	grep "pat" f	Show lines containing pattern.
43	grep "pat" f | wc -l	Count matching lines.
44	grep -c "pat" f	Same as 43, built-in count.
45	grep -h "pat" f	Suppress file name prefix.
46	grep -hi "pat" f	Case-insensitive + -h.
47	grep -hir "pat" d	Recursive, case-insensitive search.
48	grep -hin "pat" f	Show line numbers too.
49	grep -hinw "pat" f	Whole-word, case-insensitive, numbered.
50	grep -o "pat" f	Print only the matched text.
51	grep -w "pat" f	Whole-word match.
52	history	List previous commands.
53	bash script	Run file as Bash program.
54	grep "ERROR" log	All error lines.
55	grep -v "INFO" log	Invert: skip INFO lines.
56	grep -A 5 ERROR log	Error line + 5 after.
56b	grep -B 5 ERROR log	5 lines before error.
56c	grep -C 5 ERROR log	5 lines before & after.
57	sed -n '/ERROR/ p' f	Print only ERROR lines.
58	sed 's/ERROR/CRITICAL/' f	Replace first ERROR per line.

#	Command	What it does / why you care
59	`sed -ibackup 's/E/C/' f`	In-place edit with backup.
60	`sed '3 s/E/C/' f`	Replace only on line 3.
60b	`sed '3,5 s/E/C/' f`	Replace on lines 3-5.
60c	`sed -n '3,/ERROR/ p' f`	Print from line 3 to first ERROR.
61	`awk '/ERROR/{print $0}' f`	Print ERROR lines.
62	`awk '{gsub(/E/,"C"); print}' f`	Global replace E→C.
63	`awk 'BEGIN{print "HEADER"} {print} END{print "FOOTER"}' f`	Add header & footer.
64	`awk '{print $1, $2}' f`	First two columns.
65	`awk -F"," '{print $1, $2}' f`	CSV: first two fields.
66	`awk '{count[$2]++} END{print count["ERROR"]}' f`	Count occurrences in 2nd column.
67	`awk '$1>1598863888' log`	Rows whose first column > epoch.