lnav Log file navigator

START HERE

`lnav <file>`
Open a log file

i
Histogram view

/
Search with a regex

?
Built-in help

n / Shift+n
Next / previous hit

e / Shift+e
Next / previous error

:
Command prompt

;
SQL prompt

Open files formats auto-detected | **Filter and search** / and : prompts | **Mark lines** m, tags, comments | **Query with SQL** ; prompt | **Export** :write-to and friends

CONCEPTS

lnav detects log formats itself, merges all opened files by timestamp and follows them as they grow.

Piped input works too: journalctl -f | lnav. Add -t to stamp each line with its arrival time.

Prompts have history on the Up arrow; Esc cancels a prompt, Ctrl+] aborts it.

MOVING AROUND

j / Down Down a line

k / Up Up a line

Space / PgDn Down a page

b / PgUp Up a page

g / Home Top of the view

G / End Bottom of the view

Ctrl+d Down half a page

Ctrl+u Up half a page

h / Left
Scroll left half a page

l / Right
Scroll right half a page

Shift+h / Shift+l
Scroll 10 columns left / right

{ / } Previous / next section

MOVING AROUND (cont.)

Ctrl+w Toggle word wrap

` Focus the breadcrumb bar

Ctrl+x Toggle cursor mode

JUMPING BY EVENT

e / Shift+e
Next / previous error

w / Shift+w
Next / previous warning

n / Shift+n
Next / previous search hit

> / <
Next / previous hit, scrolling sideways

. / ,
Focus next / previous named search

f / Shift+f
Next / previous file

u / Shift+u
Next / previous bookmark

o / Shift+o
Next / previous message with same opid

s / Shift+s
Next / previous message-rate slowdown

F7 / F8
Previous / next breakpoint

TIME TRAVEL

d / Shift+d
Forward / back 24 hours

0 / Shift+0
Next / previous day

7 / 8 Previous / next minute

1-6
Next Nth ten-minute block of the hour

r / Shift+r
Repeat last relative goto forward / back

`:goto <time|line#|N%>`
Jump to a time, line or percentage

`:hide-lines-before <date>`
Hide lines before a date

`:hide-lines-after <date>`
Hide lines after a date

`lnav -S <time>`
Index only content since a time

SEARCH AND FILTER

/ Search lines with a regex

`:filter-in <regex>`
Show only matching lines

`:filter-out <regex>`
Hide matching lines

`:delete-filter <regex>`
Remove a filter

Tab Open filters / files panel

SEARCH AND FILTER (cont.)

Ctrl+f
Enable or disable all filters

z / Shift+z
More / less filter context lines

`:set-min-log-level <level>`
Hide messages below a level

`:filter-expr <sql-expr>`
Filter by a SQL expression

`:highlight <regex>`
Colour matching text

`:create-named-search <name> <regex>`
Save a search under a name

MARKS AND NOTES

m Mark or unmark a line

Shift+m
Mark range up to the last mark

c
Copy marked lines to clipboard

Shift+c Clear all marks

`:tag <tag>`
Tag the focused line

`:comment <text>`
Comment on the focused line

`:append-to <path>`
Append marked lines to a file

`:pipe-to <shell-cmd>`
Pipe marked lines to a command

VIEWS

q
Back to the previous view, or quit

? Show built-in help

i Toggle histogram view

t Toggle text file view

p Toggle log parser results

Shift+p
Toggle pretty-printed view

v Toggle SQL result view

Shift+t
Show elapsed time from a mark

x
Hide or show message fields

=
Pause or resume loading new data

Shift+x
Close the current file

Ctrl+l
Lo-fi mode for copying text

SQL AND EXPORT

; Open the SQL prompt

`;SELECT * FROM <log-table> LIMIT 10`
Query a log format table

`:write-csv-to <path>`
Write SQL results as CSV

SQL AND EXPORT (cont.)

`:open <path>`
Open more files or a directory

COMMAND LINE

`lnav <file> <dir>`
Open files and directories

`lnav -r <dir>`
Load files recursively

`lnav -t`
Timestamp piped input lines

`lnav -n`
Run headless, no curses UI

`lnav -c "<command>"`
Run a command, SQL or search at start

RECIPES

Follow the system journal

```
journalctl -f | lnav
# then: e to jump to errors, i
for a histogram
```

Errors from one hour as CSV

```
lnav -n -c ':goto 2026-10-12
14:00' \
  -c ';SELECT log_time,
log_body FROM all_logs WHERE
log_level = "error"' \
  -c ':write-csv-to errors.csv'
app.log
```

Trace one request

```
:filter-in req-4711
# m to mark lines, c to copy
them
# :append-to notes.txt
```