cmus Console music player

START HERE

`cmus`
Start the player

`/`
Search

`:add ~/Music`
Add music to the library

`q`
Quit (state is saved)

Enter
Play the selected track

`c`
Pause / resume

`z` / **`b`**
Previous / next track

`e`
Queue a track

Library views 1 / 2 — `y` / `e` → **Playlist / Queue** views 3 / 4 — `Enter` / `c` → **Playback** status line

CONCEPTS

cmus is modal like vi: keys act on the current view, and : opens a command line for everything else.

The queue plays first, ignores shuffle and is emptied as it plays; the library or playlist continues afterwards.

Settings, library, playlist and queue are saved on quit; :save does it manually.

VIEWS

`1` Library, tree (artist / album)

`2` Library, sorted track list

`3` Playlist

`4` Play queue

`5` File browser

Enter opens a directory, Backspace goes up, a adds the item.

`6` Filters list

`7` Key bindings and settings

Enter on a setting puts its :set command on the command line.

Tab
Move between tree columns

Backspace
Go up a directory (browser)

PLAYBACK

Enter Play selected track

`c` Pause / unpause

`x` Play from start

`v` Stop

`z` / **`b`** Previous / next track

`Z` / **`B`** Previous / next album

Left / **Right**
Seek 5 s back / forward

`h` / **`l`**
Seek 5 s back / forward

`,` / **`.`**
Seek 1 min back / forward

`=` / **`-`** Volume up / down 10%

`[` / **`]`** Volume down / up 1%

`t` Show remaining time

PLAY MODES

`s` Toggle shuffle

`r` Toggle repeat

`C` Toggle continue

When off, playback stops after each track.

`f` Toggle follow

The cursor follows the playing track.

`m`
Cycle aaa mode (all / artist / album)

PLAY MODES (cont.)

`M` Play from library / playlist

`o`
Play sorted / unsorted library

`^R` Repeat current track

LIBRARY AND QUEUE

`a` Add to library

In the browser: adds file or directory, moves down one line.

`e` Queue track

`E` Queue track at the front

`y` Add to playlist

`D` / **Delete** Remove from list

`p` / **`P`**
Move item after / before the cursor

Space
Expand artist (tree) / toggle mark

`u`
Update library cache (browser: reload)

`U` Update cache for selection

`i` Select the playing track

In the file browser, i toggles hidden files instead.

NAVIGATION

`j` / **`k`** Down / up

NAVIGATION (cont.)

`g` / **`G`** Top / bottom

`^F` / **`^B`** Page down / up

`^D` / **`^U`** Half page down / up

SEARCH AND FILTERS

`/` Search forward

All words must match; Enter ends the prompt.

`?` Search backward

`n` / **`N`** Next / previous match

`L` Live filter the current view

`F` Apply a filter expression

`:filter` *`<expr>`*
Filter the library

`:fset` *`<name>`*`=`*`<expr>`*
Save a named filter

`:factivate` *`<name>`*
Activate saved filters

Prefix a name with ! to negate it.

Filter short forms

Short	Field
`~a`	artist
`~l`	album
`~t`	title
`~g`	genre
`~y`	date (a-b range)
`~d`	duration, seconds

Combine filters with & and |, for example ~a beatles & ~y1965-1970.

COMMANDS

`:add` *`<path>`*
Add file, directory or URL

-l library, -p playlist, -q queue, -Q queue front.

`:save [file]`
Save playlist or library

-l library, -e also writes metadata.

`:set` *`<opt>`*`=`*`<val>`*
Change a setting

`:set <opt>?` shows the current value.

`:seek` *`<pos>`*
Jump in the track

Examples: +1:30, -1m, 1h.

`:vol` *`<pct>`*
Set volume

`:clear`
Clear the current view

Add -l, -p or -q to pick library, playlist or queue.

`:colorscheme` *`<name>`*
Load a colour theme

`:q`
Quit

CMUS-REMOTE

`cmus-remote -p`
Start playback

`cmus-remote -u`
Toggle pause

`cmus-remote -n / -r`
Next / previous track

`cmus-remote -s`
Stop

CMUS-REMOTE (cont.)

`cmus-remote -Q`
Print player status

`cmus-remote -v` *`<vol>`*
Change volume

Examples: +10%, -5%, 70%.

`cmus-remote -k` *`<pos>`*
Seek in the track

`cmus-remote -q` *`<file>`*
Add file to the queue

`cmus-remote -l` *`<file>`*
Add file to the library

`cmus-remote -C` *`<cmd>`*
Send any command-mode command

`cmus-remote -R / -S`
Toggle repeat / shuffle

RECIPES

First run

```
cmus
# in cmus:
:add ~/Music
# press 2 for the track list,
Enter plays
```

Play only one genre

```
:filter ~g jazz
# or keep it named
:fset jazz=genre="*jazz*"
:factivate jazz
```

Control from a keybinding or script

```
# play / pause, next, status
cmus-remote -u
cmus-remote -n
cmus-remote -Q
```