lazydocker Terminal UI for Docker

CONCEPTS

Keys depend on the focused panel: d removes a container, an image or a volume depending on where you are. Press x for the live menu.

Project Services Containers Images Volumes Networks	Logs / Stats / Env / Config / Top of the selection

Left: side panels, right: main view with tabs

Services is only filled when lazydocker runs in a directory with a docker-compose file; otherwise use Containers.

Needs access to the Docker daemon (docker group or root).

NAVIGATE

Up / **Down**
Move selection (also k / j)

Left / **Right**
Previous / next panel (also h / l)

1-6
Focus panel by number (1 projects ... 6 networks)

NAVIGATE (cont.)

[/ **]**
Previous / next tab of the main view

Enter
Focus the main panel

Esc
Return from the main panel

/ Filter the current list

PgUp / **PgDn**
Scroll the main view

Plus / **_**
Next / previous screen mode (normal, half, full)

x
Open the keybinding menu

q Quit

CONTAINERS

m
View logs (also in Project panel)

E
Exec shell in the container

r Restart

s Stop

p Pause

a Attach to the container

CONTAINERS (cont.)

e
Hide / show stopped containers

w
Open in browser (first port, assumed http)

SERVICES

u Up the selected service

S Start the service

U Up the whole project

D Down the whole project

R Restart options menu

Services also take d, s, p, r, a, m, E and w, like containers (d removes containers, not the service).

IMAGES, VOLUMES, NETWORKS

d
Remove the selected item (container, image, ...)

b
Bulk commands (prune unused ones)

IMAGES, VOLUMES, NETWORKS (cont.)

c
Run a predefined custom command (any panel)

The Images panel tabs show config and layers; Volumes and Networks show their config. Use [/] to switch.

CONFIG

o
Open the lazydocker config (Project panel)

Config lives in config.yml under the lazydocker config dir (~/.config/lazydocker on Linux). lazydocker --config prints the directory.

config.yml: custom command for containers

```
customCommands:
  containers:
    - name: bash
      attach: true
      command: 'docker exec
-it {{ .Container.ID }}
bash'
      serviceNames: []
```

CONFIG (cont.)

config.yml: tweak UI and logs

```
gui:
  scrollHeight: 2
  sidePanelWidth: 0.333
logs:
  tail: '200'
  since: '60m'
```

Custom commands show up with c on the matching panel; templates can use {{ .Container.ID }} and similar fields.

RECIPES

Tail logs of a crashing container

```
# 3 to focus Containers,
pick it
m         # logs tab
/         # filter if needed
r         # restart and watch
```

Shell into a service

```
# in a compose project dir
lazydocker
2         # Services panel
E         # exec shell
```

Free disk space

```
4         # Images panel
b         # bulk commands:
prune images
5         # Volumes
b         # prune unused
volumes
```

Bounce a stack

```
lazydocker
D         # down project
U         # up project
```