# gum Glamorous shell scripts

## CONCEPTS

gum prints the user's answer to stdout, so capture it with $(...) and keep prompts on the terminal.

**gum prompt** choose, input, confirm
↓ stdout / exit code
**Shell variable** $(...) or if
↓ your logic
**Script action** spin, log, style

Every flag has an env var GUM_<COMMAND>_<OPTION>; flags override it. Set them once for a consistent look.

## CHOOSE AND FILTER

**gum choose** *<a> <b> <c>*
Pick one from a list

**gum choose --no-limit**
Pick any number (stdin or args)

**gum choose --limit** *<n>*
Cap the number of picks

**gum choose --header** *<text>*
Title above the list

**gum choose --height** *<n>*
Visible rows of the list

**gum filter**
Fuzzy search lines from stdin

**gum filter --limit** *<n>*
Allow several matches

**gum filter --placeholder** *<text>*
Hint in the search box

**gum file** *<dir>*
Browse and pick a file

**gum table** < *<file>*.csv
Pick a row from CSV

## INPUT

**gum input**
Read one line of text

## INPUT (cont.)

**gum input --placeholder** *<text>*
Show a hint when empty

**gum input --value** *<text>*
Prefill the field

**gum input --password**
Hide typed characters

**gum input --prompt** *<text>*
Change the prompt marker

**gum write**
Multi-line text, Ctrl+D to finish

## CONFIRM AND RUN

**gum confirm**
Ask yes/no, exit 0 or 1

**gum confirm && *<cmd>* || *<cmd>***
Branch on the answer

**gum confirm "*<question>*"**
Custom question text

**gum spin --title** *<text>* **--** *<cmd>*
Spinner while a command runs

**gum spin --show-output --** *<cmd>*
Print command output after

## STYLE AND OUTPUT

**gum style** *<text>*
Print styled text

**gum style --foreground** *<n> <text>*
Set text colour (0-255 or hex)

**gum style --border double** *<text>*
Draw a border

**gum format** *<markdown>*
Render markdown

**gum format -t code**
Syntax-highlight code from stdin

**gum pager** < *<file>*
Scroll long text

**gum log --level** *<lvl> <msg>*
Log with a level