man Manual page lookup

START HERE

`man <command>`
Read a manual page

`n / N`
Next / previous match

`man -k <word>`
Search page descriptions

`q`
Quit the page

`man <n> <name>`
Pick a section

`<command> --help`
Quick usage summary

`/<text>`
Search inside a page

`help <builtin>`
Help for a shell builtin

Do not know the name a topic or word → `man -k` → **Pick a hit** name (section) → `man` → **Read the page** in the pager → `/ search` → **Find the option** or example

CONCEPTS

A name can exist in several sections: printf(1) is the command, printf(3) the C function. Write man 3 printf to pick.

Manual sections

Section	Holds
1	User commands
2	System calls
3	Library functions
4	Devices and special files
5	File formats and config
6	Games
7	Overviews and conventions
8	Admin commands

man shows pages through a pager, normally less, so every less key works inside it.

Shell builtins (cd, export, alias) have no own page: use help <name>, or man bash and search there.

OPENING PAGES

`man <name>`
Show the manual page

`man <section> <name>`
Show the page from one section

`man -a <name>`
Show every matching page in turn
Press q to move on to the next section's page.

`man -l <file>`
Open a page from a local file

`man -w <name>`
Print the path of the page file

`man -aw <name>`
Print the paths of all matching pages

`man <name> <sub>`
Open subcommand page (git-commit)
man git commit finds git-commit(1); --no-subpages turns this off.

OPENING PAGES (cont.)

`man -P <pager> <name>`
Use another pager for one run

`man -P cat <name>`
Dump the page without a pager

`man -I <name>`
Match the name case-sensitively

`MANWIDTH=80 man <name>`
Format the page to a set width

SEARCHING PAGES

`man -k <word>`
Search names and descriptions

`apropos <word>`
Same as man -k

`man -f <name>`
One-line summary of a page

`whatis <name>`
Same as man -f

`man -k <word> -s <n>`
Limit a search to one section

`man -k '^<prefix>'`
Search with a regex on the names

`man -k --names-only <word>`
Match page names only

`man -k --wildcard '<glob>'`
Search names with a wildcard

`man -K <text>`
Search the full text of all pages
Slow: it opens every page. Combine with -s to narrow it.

`mandb`
Rebuild the index behind -k and -f
New pages are invisible to -k until the database is updated (usually needs root).

INSIDE THE PAGER

`Space / b`
Page forward / back

INSIDE THE PAGER (cont.)

`j / k`
Line down / up

`g / G`
Top / bottom of the page

`/<text>`
Search forward

`?<text>`
Search backward

`n / N`
Next / previous match

`&<text>`
Show only lines matching

`h`
Pager key help

`q`
Quit

`/^ *-<x>`
Jump to the description of option -x
*The pattern matches an option at the start of a line, e.g. /^ *-r.*

`/^SYNOPSIS`
Jump to a section heading

`MANPAGER='less -R +Gg'`
Choose the pager for man

OTHER HELP SOURCES

`<command> --help`
Built-in usage summary

`help <builtin>`
Help for a shell builtin

`help`
List all shell builtins

`type <name>`
Is it a builtin, alias or file?

`which <name>`
Path of the executable

`info <topic>`
Open the GNU Info manual

`ls /usr/share/doc/<pkg>`
Package docs and examples

OTHER HELP SOURCES (cont.)

`tldr <command>`
Short community examples (extra tool)

`man man`
The manual page for man itself

`man 7 man-pages`
How pages are written and laid out

`compgen -b`
List builtins for completion

READING A PAGE

Conventions in SYNOPSIS

Written	Means
`[x]`	optional
`<x> or x in italics`	replace with a value
`a | b`	pick one
`x ...`	can repeat

Read NAME and SYNOPSIS first, then jump to the option you need; EXAMPLES and SEE ALSO often save the most time.

Not every page has EXAMPLES. If it is thin, try the --help output or a tldr page.

RECIPES

Find the right command and read it

```
man -k 'copy file'
man -f cp
man 1 cp
# /^ *-r then n
```

Which page, which section?

```
man -aw printf
man -f printf
man 3 printf
```

Save a readable plain-text page

```
man ls | col -bx > ls.txt
# or: MANWIDTH=80 man -P cat ls
> ls.txt
```

Look up a flag without leaving the shell

```
man -P cat tar | grep -n -- '--
exclude'
tar --help | grep -- --exclude
```