# find Search the file tree

## CONCEPTS

find walks every path you give it and tests each file against the expression, left to right. Quote patterns so the shell does not expand them.

**Start paths** where to look
↓ options
**Options** depth, symlinks
↓ tests
**Tests** name, type, size, time
↓ actions
**Actions** print, exec, delete

Without an action, find prints matches (-print). Test a command with -print before you switch to -delete or -exec.

Examples follow GNU findutils 4.10. BSD and macOS find lack -printf, -regextype and -type f,d, and use find -E for extended regex.

## NAME & TYPE

`find <dir> -name '<glob>'`
Match the file name

`find <dir> -iname '<glob>'`
Match name, ignore case

`find <dir> -type f`
Regular files only

`find <dir> -type d`
Directories only

`find <dir> -type l`
Symbolic links only

`find <dir> -path '<glob>'`
Match the whole path

`find <dir> -regex '<regex>'`
Regex on the whole path

`find <dir> -xtype l`
Broken symlinks

## SIZE & TIME

`find <dir> -size +<n>M`
Larger than n MiB

`find <dir> -size -<n>k`
Smaller than n KiB

## SIZE & TIME (cont.)

`find <dir> -empty`
Empty files and directories

`find <dir> -mtime -<n>`
Modified in the last n days

`find <dir> -mtime +<n>`
Modified more than n days ago

`find <dir> -mmin -<n>`
Modified in the last n minutes

`find <dir> -newer <file>`
Newer than a reference file

`find <dir> -newermt '<date>'`
Modified after a date

## OWNER & PERMISSIONS

`find <dir> -user <name>`
Owned by a user

`find <dir> -group <name>`
Owned by a group

`find <dir> -perm 644`
Exactly these mode bits

`find <dir> -perm -u+x`
All listed bits set

`find <dir> -perm /022`
Any listed bit set

## DEPTH & PRUNING

`find <dir> -maxdepth <n>`
Descend at most n levels

`find <dir> -mindepth <n>`
Ignore the top n levels

`find <dir> -name <d> -prune -o -print`
Skip a directory tree

`find <dir> -xdev`
Stay on one filesystem

## COMBINING

`find <dir> -name '*.c' -o -name '*.h'`
Either pattern (OR)

`find <dir> ! -name '<glob>'`
Negate a test

`find <dir> \( <a> -o <b> \) <c>`
Group with parentheses

## ACTIONS

`find <dir> -print`
Print matches (default)

`find <dir> -print0`
Print NUL-separated names

`find <dir> -exec <cmd> {} \;`
Run command once per file

`find <dir> -exec <cmd> {} +`
Batch files per command

`find <dir> -ok <cmd> {} \;`
Ask before each run

`find <dir> -execdir <cmd> {} \;`
Run inside the file's directory

`find <dir> -delete`
Delete matches

`find <dir> -printf '<fmt>'`
Custom output format

## XARGS

`find <dir> -print0 | xargs -0 <cmd>`
Pipe names safely

`xargs -0 -n <n> <cmd>`
At most n names per run

`xargs -0 -P <n> <cmd>`
Run n jobs in parallel

`xargs -0 -I {} <cmd> {}`
Place name inside a command

## RECIPES

### Delete old logs

```
find /var/log/app -name '*.log' -mtime +30 -print
# check the list, then swap in -delete
find /var/log/app -name '*.log' -mtime +30 -delete
```

### Replace text in many files

```
find . -name '*.md' -print0 \
  | xargs -0 sed -i 's/<old>/<new>/g'
```

### Grep, skipping .git

```
find . -name .git -prune -o -type f \
  -exec grep -l '<text>' {} +
```

### Ten biggest files

```
find . -type f -printf '%s %p\n' \
  | sort -nr | head
```