wget Download and mirror

Fetch URL retries, redirects → saves → **Local file** -O, -P, -nd name it → -r parses HTML → **Follow links** -l depth, -np, -A/-R → -k rewrites → **Offline copy** -m -k -p

CONCEPTS

wget saves to a file by default and keeps going on its own: it retries, follows redirects and can recurse. curl prints to stdout and does not recurse; see the curl sheet for APIs.

An existing file is not overwritten: wget saves as name.1, name.2. Use -c to resume, -N to refresh only newer files, or -nc to skip.

DOWNLOAD

`wget <url>`
Download to the current directory

`wget -O <file> <url>`
Save under a chosen name

`wget -P <dir> <url>`
Save into a directory

`wget -c <url>`
Resume an interrupted download

`wget -i <file>`
Read URLs from a file

`wget -q <url>`
Quiet, no output

DOWNLOAD (cont.)

`wget -nv <url>`
Short one-line output

`wget -b <url>`
Run in the background

`wget -O - <url>`
Write the body to stdout

`wget --show-progress -q <url>`
Quiet but keep the progress bar

`wget -nc <url>`
Skip if the file already exists

`wget -N <url>`
Only fetch if remote is newer

`wget --content-disposition <url>`
Use the server-suggested filename

RELIABILITY AND SPEED

`wget -t <n> <url>`
Number of retries (0 is unlimited)

`wget --retry-connrefused <url>`
Retry on refused connections

`wget --retry-on-http-error=503,429 <url>`
Retry on listed HTTP codes

RELIABILITY AND SPEED (cont.)

`wget -T <sec> <url>`
Set all timeouts

`wget --limit-rate=<rate> <url>`
Cap bandwidth (e.g. 500k, 2m)

`wget -w <sec> -i <file>`
Pause between downloads

RECURSIVE AND MIRROR

`wget -r <url>`
Follow links recursively (depth 5)

`wget -r -l <n> <url>`
Set recursion depth (0 or inf is unlimited)

`wget -np -r <url>`
Never ascend to the parent directory

`wget -m <url>`
Mirror: -r -N -l inf

`wget -k <url>`
Convert links for offline viewing

`wget -p <url>`
Also fetch images, CSS, scripts

`wget -nd -r <url>`
Flatten into one directory

RECURSIVE AND MIRROR (cont.)

`wget -nH --cut-dirs=<n> -r <url>`
Drop host and N path levels

`wget -A <list> -r <url>`
Accept only these extensions

`wget -R <list> -r <url>`
Reject these extensions

`wget --accept-regex=<re> -r <url>`
Accept URLs matching a regex

`wget --reject-regex=<re> -r <url>`
Skip URLs matching a regex

`wget -H -D <domains> -r <url>`
Span hosts but only these domains

`wget -E -k -p <url>`
Save one page fully for offline reading

`wget -e robots=off -r <url>`
Ignore robots.txt

HTTP, AUTH AND HEADERS

`wget --header='<Name: value>' <url>`
Add a request header

HTTP, AUTH AND HEADERS (cont.)

`wget --user=<u> --ask-password <url>`
Basic auth, prompt for password

`wget --load-cookies <file> <url>`
Send cookies from a file

`wget -U '<agent>' <url>`
Set the User-Agent

`wget -S <url>`
Show the server response headers

`wget --spider <url>`
Check a URL without saving it

TLS AND LOGGING

`wget --no-check-certificate <url>`
Skip certificate validation

`wget -o <file> <url>`
Write the log to a file

RECIPES

Resume with retries

```
wget -c -t 0 --retry-connrefused -T 30 <url>
```

Grab only PDFs from a page

```
wget -r -l 1 -nd -np -A pdf -P pdfs <url>
```