

# find examples

General note to `-exec {} +` vs `-exec {} \;`: The following executes program and sends a space separated list of quoted filenames to it. Program may be called multiple times if the amount of arguments exceeds `ARG_MAX`.

```
find . -exec program {} +
```

The following executes program with one! quoted filename, then executes program again with the next etc.

```
find . -exec program {} \;
```

Find and extract all rar files in subfolders into the current folder:

```
find . -name *.rar -exec unrar x {} \;
```

Find duplicate files:

```
find . ! -empty -type f -exec md5sum {} + | sort | uniq -w32 -dD
```

Find duplicate filenames:

```
find . -mindepth 1 -printf '%h %f\n' | sort -t ' ' -k 2,2 | uniq -f 1 --all-repeated=separate | tr ' ' '/'
```

Find files older than 7 years:

```
find . -mtime +2555 -print
```

Find directories containing files older than 30 days:

```
find . -type f -mtime +30 -printf '%h\n' | sort | uniq > old.txt
```

Find directories containing ONLY files older than 30 days (not working with subdirs properly):

```
find . -type f -mtime +30 -printf '%h\n' | sort | uniq > old.txt  
find . -type f -mtime -30 -printf '%h\n' | sort | uniq > new.txt  
grep -vf new.txt old.txt
```

Remove all old temporary MS Office lock files:

```
find . -type f -mtime +10 -name "~*" -print0 | xargs -r0 rm --
```

Show top level folders and full file count and file count>7years:

```
find . -maxdepth 1 -type d -print0 -exec echo -n " " \; -exec sh -c "find  
'{}' -type f | wc -l | tr -d '\n'" \; -exec echo -n " " \; -exec sh -c "find
```

```
'{}' -type f -mtime +2555 | wc -l" \;
```

Convert tabs to spaces at the beginning of lines in source code:

```
find . -iname '*.php' -type f -exec bash -c 'expand -i -t 4 "$0" > /tmp/e && mv /tmp/e "$0"' {} \;
```

Better with 'sponge' from the moreutils package as it retains permissions:

```
find . -iname '*.php' -type f -exec bash -c 'expand -i -t 4 "$0" | sponge "$0"' {} \;
```

Fixing permissions of directories / files of a path:

```
find /path/to/base/dir -type d -print0 | xargs -0 chmod 755
find /path/to/base/dir -type f -print0 | xargs -0 chmod 644
```

Recursive file renaming (e.g. from "something [xxxxp].ext" to "something.xxxxp.ext") :

```
#NOTE: use -n in rename parameter to test!
find . -type f -name "*\[*" -print0 | xargs -r0 rename 's/(.*)
\[([d\d\d\d?])\](.*)/$1.$2$3/' --
```

Recursive file renaming extension (e.g. from "something.oldext" to "something.newext") :

```
find . -type f -name '*.oldext' -print0 | xargs -0 rename
's/\.oldext/\.newext/'
```

Move files in subdirectories to other directory while retaining directory structure

```
find . -maxdepth 2 -type f -print | cpio -pvduB
/media/music/_OrganizeA_Leftover ; find . -maxdepth 2 -type f -delete

#Do this for each directory
for d in */; do
    cd $d; find . -maxdepth 2 -type f -print | cpio -pvduB
/media/music/_OrganizeA_Leftover ; find . -maxdepth 2 -type f -delete; cd ..
done

#If directory must not have trailing /, use this in substitution
${d%/*}

delete empty directories:
find /home/something/ -type d -empty -delete
```

get the total filesize of specific find command:

```
find . -type f -not -iname "*.epub" -not -iname "*.mobi" -print0 | du --
files0-from=- -hc | tail -n1
84M total
```

```
find . -type f -not -iname "*.epub" -not -iname "*.mobi" -exec du {} \; |  
awk '{ total += $1 }END{ print total }'  
85864
```

## Replace spaces with underscores

Mass replace spaces with underscores

```
for file in *; do mv "$file" `echo $file | tr ' ' '_'` ; done
```

## Find largest subdirectories

```
du -hd1 | sort -h
```

## Recursively changing extension

```
find . -type f -name "*.part" -exec sh -c 'mv "$1" "${1%.part}!.qB"' _ {} \;  
find . -type f -name '*.JPG' -print0 | xargs -0 sudo rename 's/.JPG$/ .jpg/'
```

## Recursively delete empty directories

```
find . -type d -empty -delete
```

From:

<http://wuff.dyndns.org/> - **Wulf's Various Things**

Permanent link:

<http://wuff.dyndns.org/doku.php?id=linux:find>

Last update: **2026/06/10 09:30**

