

[Pro](#) [Teams](#) [Pricing](#) [Documentation](#)**npm**

Search

**copyfiles** [DT](#)2.4.1 • [Public](#) • Published 2 years ago [Readme](#) [Code](#) [Beta](#) [7 Dependencies](#) [882 Dependents](#) [18 Versions](#)**copyfiles** [build](#) [passing](#)

copy files easily

Install

```
npm install copyfiles -g
```

Command Line

```
Usage: copyfiles [options] inFile [more files ...] outDirectory
```

Options:

- u, --up slice a path off the bottom of the paths
- a, --all include files & directories begining with a dot (.
- f, --flat flatten the output
- e, --exclude pattern or glob to exclude (may be passed multiple
- E, --error throw error **if** nothing is copied
- V, --verbose print more information to console
- s, --soft **do** not overwrite destination files **if** they exist
- F, --follow follow symbolink links
- v, --version Show version number
- h, --help Show [help](#)

copy some files, give it a bunch of arguments, (which can include globs), the last one is the out directory (which it will create if necessary). Note: on windows globs must be **double quoted**, everybody else can quote however they please.

```
copyfiles foo foobar foo/bar/*.js out
```

you now have a directory called out, with the files foo and foobar in it, it also has a directory named foo with a directory named bar in it that has all the files from foo/bar that match the glob.

If all the files are in a folder that you don't want in the path out path, ex:

```
copyfiles something/*.js out
```

which would put all the js files in out/something , you can use the --up (or -u ) option

```
copyfiles -u 1 something/*.js out
```

which would put all the js files in out

you can also just do -f which will flatten all the output into one directory, so with files ./foo/a.txt and ./foo/bar/b.txt

```
copyfiles -f ./foo/*.txt ./foo/bar/*.txt out
```

will put a.txt and b.txt into out

if your terminal doesn't support globstars then you can quote them

```
copyfiles -f ./foo/**/*.txt out
```

does not work by default on a mac

but

```
copyfiles -f "./foo/**/*.txt" out
```

does.

You could quote globstars as a part of input:

```
copyfiles some.json "./some_folder/*.json" ./dist/ && echo 'JSON file'
```

You can use the -e option to exclude some files from the pattern, so to exclude all files ending in .test.js you could do

```
copyfiles -e "**/*.test.js" -f ./foo/**/*.js out
```

Other options include

- `-a` or `--all` which includes files that start with a dot.
- `-s` or `--soft` to soft copy, which will not overwrite existing files.
- `-F` or `--follow` which follows symbolinks

## copyup

also creates a `copyup` command which is identical to `copyfiles` but `-up` defaults to 1

### Programic API

```
var copyfiles = require('copyfiles');
```

```
copyfiles([paths], opt, callback);
```

takes an array of paths, last one is the destination path, also takes an optional argument which the `-u` option if a number, otherwise if it's `true` it's the flat option or if it is an object it is a hash of the various options (the long version e.g. `up`, `all`, `flat`, `exclude`, `error`, `verbose`, `follow`, and `soft`)

### Tilde support for home directory

when the `src/dest` path start with tilde for home directory under windows, please make sure `-u` or `-f` is added in options or use `copyup` command. if not you will get `Error: Illegal characters in path.`

### Keywords

none  
Install

```
> npm i copyfiles
```

### Repository

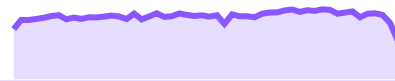
 [github.com/calvinmetcalf/copyfiles](https://github.com/calvinmetcalf/copyfiles)

### Homepage

 [github.com/calvinmetcalf/copyfiles#readme](https://github.com/calvinmetcalf/copyfiles#readme)

### Weekly Downloads

416,942



Version

2.4.1

License

MIT

Unpacked Size

22.9 kB

Total Files

8

Issues

39

Pull Requests

7

Last publish

2 years ago

Collaborators

[➤ Try on RunKit](#)[🚩 Report malware](#)

## Support

[Help](#)[Advisories](#)[Status](#)[Contact npm](#)

## Company

[About](#)[Blog](#)[Press](#)

## Terms & Policies

[Policies](#)[Terms of Use](#)[Code of Conduct](#)[Privacy](#)