

Move a module from devDependencies to dependencies in npm package.json

Asked 2 years ago Active 3 months ago Viewed 22k times



Is there any short command to move a module from `devDependencies` to `dependencies` in `package.json`?

113

I find myself always doing this:



```
npm uninstall <module_name> --save-dev  
npm install <module_name> --save
```



15

Is there a shorter approach to this?

npm

package.json

edited Jan 2 at 8:54



Shakespeare

1,085 3 14 21

asked Oct 24 '17 at 5:46



Emad Emami

1,877 2 15 32

5 Answers



Yes! to move a module from `devDependencies` to `dependencies` :

123

```
npm install <module_name> --save-prod
```



edited Jan 2 at 10:44



Shakespeare

1,085 3 14 21

answered Dec 22 '17 at 8:31



Francois Wouts

2,005 1 13 11



If you want to do the opposite (i.e. move a module from `dependencies` to `devDependencies`) just do:

132

```
npm install <module_name> --save-dev
```

or shorthand:

```
npm i <module_name> -D
```

shorthand to move from devDependencies to dependencies (prod):

```
npm i <module_name> -P
```

edited Jan 2 at 11:20



Shakespear

1,085 3 14 21

answered Jan 15 '18 at 9:17



Sid

7,549 6 29 41

Does not work with npm v3 though. – [Gert Hengeveld](#) Jan 17 '18 at 7:19 ✎

5 Works for me, v5.6 – [Legends](#) Mar 12 '18 at 17:02

4 seems like it is working 6.0.0 as well – [Sid](#) May 1 '18 at 17:44

1 shorthand `npm i module-example -D` will also work to move from prod to dev, as well as `npm i module-example -P` to move from dev to prod dependencies – [Andriy](#) Nov 4 '18 at 9:25

@Andriy thanks, didn't thought about it. Will update the answer! – [Sid](#) Nov 5 '18 at 13:34

In yarn:

10

move a module from devDependencies to dependencies :

```
yarn remove <module_name> --dev&yarn add <module_name>
```

move a module from dependencies to devDependencies :

```
yarn remove <module_name>&yarn add <module_name> --dev
```

As said in the comments, the command actually deletes the module and reinstall it in the new place.

edited Jun 24 at 13:40

answered Feb 3 at 10:01



yohaiz

309 2 10

1 Is this really right? Is there no way in yarn to move things from `dependencies` to `devDependencies` without removing and re-adding? – LondonRob Feb 12 at 10:23

@LondonRob that's the only way i've found – yohaiz Feb 12 at 16:24

Won't there be a race condition running in background like that? – seeker_of_bacon Apr 8 at 21:05 ✎

5 This is not moving, This is simply deleting the package and installing it again – Thanveer Shah Apr 26 at 13:18

@ThanveerShah you right, i added clarification – yohaiz Jun 24 at 13:51



0



The problem with using `npm` or `yarn` commands is that there is a chance that **the version that is re-added is a different version than the one that is currently used**. If this is what you want - **both** a move and an upgrade - then go ahead and use the accepted answer.

If not, simply **manually edit your** `package.json` to move the line from the `devDependencies` object to the `dependencies` object (creating it if necessary). You can go the other direction too.

~~The lock file doesn't hold any information about if things are prod or dev dependencies, so that doesn't need to be updated.~~ You can do a `npm / yarn install` afterwards to fix up any flags in the lock files.

edited Jul 31 at 2:09

answered Jul 15 at 1:48



eedrah

596 4 23

While there is only a single `dependencies` top-level section in `package-lock.json`, each entry potentially has a `dev` flag to indicate whether it is a dev dependency or a transitive dependency of one. See docs.npmjs.com/files/package-lock.json#dev "If [dev is] true then this dependency is either a development dependency ONLY of the top level module or a transitive dependency of one. This is false for dependencies that are both a development dependency of the top level and a transitive dependency of a non-development dependency of the top level." – rob3c Jul 23 at 0:53



0

If your project doesn't have a lockfile or a shrinkwrap file yet, you can simply move the corresponding line in your `package.json`.

(I'm not recommending not using lockfiles)

11/12/2019

Move a module from devDependencies to dependencies in npm package.json - Stack Overflow

answered Feb 3 at 9:33



sean

452 6 11