

npm-update

Update a package

SYNOPSIS

```
npm update [-g] [<pkg>...]
```

```
aliases: up, upgrade
```

DESCRIPTION

This command will update all the packages listed to the latest version (specified by the `tag` config), respecting semver.

It will also install missing packages. As with all commands that install packages, the `--dev` flag will cause `devDependencies` to be processed as well.

If the `-g` flag is specified, this command will update globally installed packages.

If no package name is specified, all packages in the specified location (global or local) will be updated.

As of `npm@2.6.1`, the `npm update` will only inspect top-level packages. Prior versions of `npm` would also recursively inspect all dependencies. To get the old behavior, use `npm --depth 9999 update`.

As of `npm@5.0.0`, the `npm update` will change `package.json` to save the new version as the minimum required dependency. To get the old behavior, use `npm update --no-save`.

EXAMPLES

IMPORTANT VERSION NOTE: these examples assume `npm@2.6.1` or later. For older versions of `npm`, you must specify `--depth 0` to get the behavior described below.

For the examples below, assume that the current package is `app` and it depends on dependencies, `dep1` (`dep2`, .. etc.). The published versions of `dep1` are:

```
{
  "dist-tags": { "latest": "1.2.2" },
  "versions": [
    "1.2.2",
    "1.2.1",
    "1.2.0",
    "1.1.2",
    "1.1.1",
    "1.0.0",
    "0.4.1",
    "0.4.0",
    "0.2.0"
  ]
}
```

Caret Dependencies

If `app`'s `package.json` contains:

```
"dependencies": {
  "dep1": "^1.1.1"
}
```

Then `npm update` will install `dep1@1.2.2`,
because `1.2.2` is `latest` and `1.2.2` satisfies `^1.1.1`.

Tilde Dependencies

However, if `app`'s `package.json` contains:

```
"dependencies": {
  "dep1": "~1.1.1"
}
```

In this case, running `npm update` will install `dep1@1.1.2`. Even though the `latest` tag points to `1.2.2`, this version does not satisfy `~1.1.1`, which is equivalent to `>=1.1.1 <1.2.0`. So the highest-sorting version that satisfies `~1.1.1` is used, which is `1.1.2`.

Caret Dependencies below 1.0.0

Suppose `app` has a caret dependency on a version below `1.0.0`, for example:

```
"dependencies": {  
  "dep1": "^0.2.0"  
}
```

`npm update` will install `dep1@0.2.0` , because there are no other versions which satisfy `^0.2.0` .

If the dependence were on `^0.4.0` :

```
"dependencies": {  
  "dep1": "^0.4.0"  
}
```

Then `npm update` will install `dep1@0.4.1` , because that is the highest-sorting version that satisfies `^0.4.0` (`>= 0.4.0 <0.5.0`)

Updating Globally-Installed Packages

`npm update -g` will apply the `update` action to each globally installed package that is `outdated` – that is, has a version that is different from `latest` .

NOTE: If a package has been upgraded to a version newer than `latest` , it will be *downgraded*.