

NAME**npm-explain** – Explain installed packages**Synopsis**

npm explain <folder | specifier>

alias: why

Description

This command will print the chain of dependencies causing a given package to be installed in the current project.

Positional arguments can be either folders within **node_modules**, or **name@version-range** specifiers, which will select the dependency relationships to explain.

For example, running **npm explain glob** within npm's source tree will show:

```
glob@7.1.6
node_modules/glob
  glob@"^7.1.4" from the root project
```

```
glob@7.1.1 dev
node_modules/tacks/node_modules/glob
  glob@"^7.0.5" from rimraf@2.6.2
node_modules/tacks/node_modules/rimraf
  rimraf@"^2.6.2" from tacks@1.3.0
node_modules/tacks
  dev tacks@"^1.3.0" from the root project
```

To explain just the package residing at a specific folder, pass that as the argument to the command. This can be useful when trying to figure out exactly why a given dependency is being duplicated to satisfy conflicting version requirements within the project.

```
$ npm explain node_modules/nyc/node_modules/find-up
find-up@3.0.0 dev
node_modules/nyc/node_modules/find-up
  find-up@"^3.0.0" from nyc@14.1.1
node_modules/nyc
  nyc@"^14.1.1" from tap@14.10.8
node_modules/tap
  dev tap@"^14.10.8" from the root project
```

Configuration

```
<!-- AUTOGENERATED CONFIG DESCRIPTIONS START --> <!-- automatically generated, do not
edit manually --> <!-- see lib/utils/config/definitions.js -->
```

json

- Default: false
- Type: Boolean

Whether or not to output JSON data, rather than the normal output.

- In **npm pkg set** it enables parsing set values with `JSON.parse()` before saving them to your **package.json**.

Not supported by all npm commands. <!-- automatically generated, do not edit manually --> <!-- see lib/utils/config/definitions.js -->

workspace

- Default:
- Type: String (can be set multiple times)

Enable running a command in the context of the configured workspaces of the current project while filtering by running only the workspaces defined by this configuration option.

Valid values for the **workspace** config are either:

- Workspace names
- Path to a workspace directory
- Path to a parent workspace directory (will result in selecting all workspaces within that folder)

When set for the **npm init** command, this may be set to the folder of a workspace which does not yet exist, to create the folder and set it up as a brand new workspace within the project.

This value is not exported to the environment for child processes. <!-- automatically generated, do not edit manually --> <!-- see lib/utils/config/definitions.js -->

<!-- AUTOGENERATED CONFIG DESCRIPTIONS END -->

See Also

- npm help config
- npm help npmrc
- npm help folders
- npm help ls
- npm help install
- npm help link
- npm help prune
- npm help outdated
- npm help update