npm-dist-tag

Modify package distribution tags

SYNOPSIS

```
npm dist-tag add <pkg>@<version> [<tag>]
npm dist-tag rm <pkg> <tag>
npm dist-tag ls [<pkg>]

aliases: dist-tags
```

DESCRIPTION

Add, remove, and enumerate distribution tags on a package:

- add: Tags the specified version of the package with the specified tag, or the `--tag` config if not specified. If you have two-factor authentication on auth-and-writes then you'll need to include a one-time password on the command line with `--otp <one-time password>`.
- rm: Clear a tag that is no longer in use from the package.
- ls: Show all of the dist-tags for a package, defaulting to the package in the current prefix.

This is the default action if none is specified.

A tag can be used when installing packages as a reference to a version instead of using a specific version number:

```
npm install <name>@<tag>
```

When installing dependencies, a preferred tagged version may be specified:

```
npm install --tag <tag>
```

This also applies to **`npm dedupe`**.

Publishing a package sets the **`latest`** tag to the published version unless the **`--tag`** option is used. For example, **`npm publish --tag=beta`**.

By default, **`npm install <pkg>`** (without any **`@<version>`** or **`@<tag>`** specifier) installs the **`latest`** tag.

PURPOSE

Tags can be used to provide an alias instead of version numbers.

For example, a project might choose to have multiple streams of development and use a different tag for each stream, e.g., **`stable`**, **`beta`**, **`dev`**, **`canary`**.

By default, the **`latest`** tag is used by npm to identify the current version of a package, and **`npm install <pkg>`** (without

any `@<version>` or `@<tag>` specifier) installs the `latest` tag. Typically, projects only use the `latest` tag for stable release versions, and use other tags for unstable versions such as prereleases.

The `next` tag is used by some projects to identify the upcoming version.

By default, other than `latest`, no tag has any special significance to npm itself.

CAVEATS

This command used to be known as `npm tag`, which only created new tags, and so had a different syntax.

Tags must share a namespace with version numbers, because they are specified in the same slot: `npm install <pkg>@<version>` vs `npm install <pkg>@<tag>`.

Tags that can be interpreted as valid semver ranges will be rejected. For example, `v1.4` cannot be used as a tag, because it is interpreted by semver as `>=1.4.0 <1.5.0`.
See https://github.com/npm/npm/issues/6082.

The simplest way to avoid semver problems with tags is to use tags that do not begin with a number or the letter `v`.