

NPM Cheatsheet

A comprehensive cheat sheet for Node Package Manager (NPM), covering essential commands and functionalities for managing JavaScript packages and dependencies.



Basic NPM Commands

Installation

npm install or npm i	Installs all dependencies defined in package.json in the current directory. npm install
<pre>npm install <package_name></package_name></pre>	Installs a specific package and adds it as a dependency to <pre>package.json</pre> <pre>npm install lodash</pre>
<pre>npm install <package_name>save-dev or -D</package_name></pre>	Installs a package as a development dependency. Useful for tools like testing frameworks. npm install jestsave-dev
npm install <package_name>global or -g</package_name>	Installs a package globally, making it available in the system's PATH. Often used for command-line tools. npm install create-react-appglobal
<pre>npm install <git_repo></git_repo></pre>	Install a package directly from a git repository. npm install git+ssh://git@github.com:user/repo.git
npm ci	Install dependencies with a clean slate, based on package-lock.json. Faster and ensures consistency across environments.

Updating Packages

npm update	Updates all packages listed in package.json to the latest version that satisfies the version ranges specified. npm update
npm update <package_name></package_name>	Updates a specific package to the latest version that satisfies its version range. npm_update_lodash
npm audit fix	Attempts to automatically fix known security vulnerabilities in your dependencies. npm audit fix
npm outdated	Lists outdated packages. npm outdated

Uninstalling Packages

npm uninstall <package_name></package_name>	Uninstalls a package from the <pre>node_modules</pre> directory and removes it from <pre>package.json</pre> dependencies.
	npm uninstall lodash
<pre>npm uninstall <package_name>save-dev</package_name></pre>	Uninstalls a package from development dependencies.
	npm uninstall jestsave-dev
npm uninstall -g <package_name></package_name>	Uninstalls a globally installed package.
	npm uninstall -g create-react-app
npm clean-install	Remove node_modules and install dependencies.
	npm clean-install

Managing Dependencies

Package Information

npm view <package_name></package_name>	Displays detailed information about a package from the npm registry.
	npm view lodash
npm view <package_name> version</package_name>	Shows the latest version of a specific package.
	npm view lodash version

Page 1 of 3 https://cheatsheetshero.com

npm view <package_name> dependencies</package_name>	Lists the dependencies of a specific package.
	npm view lodash dependencies
(npm 1s	Lists installed packages and their dependencies in a tree structure.
	npm ls

Working with package.json

npm init	Guides you through creating a new package.json file for your project.
	npm init
npm init -y or npm inityes	Creates a package.json file with default values without prompting for input.
	npm init -y
npm installproduction	Installs only production dependencies (excluding devDependencies) from package.json.
	npm installproduction
npm pruneproduction	Removes extraneous packages. If theproduction flag is set, only removes the packages that are not listed in dependencies.
	npm pruneproduction

Scripts

<pre>npm run <script_name></script_name></pre>	Executes a script defined in the scripts section of package.json . npm run start
npm start	Runs the start script defined in package.json. Often used to start the application. npm start
npm test	Runs the test script defined in package.json. Used for running tests. npm test
npm stop	Runs the stop script defined in package.json. Used for stopping the application. npm stop
npm restart	Runs the restart script defined in package.json. Used for restarting the application. npm restart

Advanced NPM Usage

Publishing Packages

npm login	Logs you into the npm registry using your credentials.
	npm login
npm publish	Publishes the package in the current directory to the npm registry. Make sure to increment the version number in package.json before publishing.
	npm publish
npm unpublish	Removes a published package from the npm registry. Use with caution!
<pre><package_name>@<version></version></package_name></pre>	npm unpublish my-package@1.0.0
npm owner add <user> <package_name></package_name></user>	Adds a user as an owner of a package, allowing them to manage the package.
	npm owner add username my-package

Managing Cache

npm cache cleanforce	Clears the npm cache. Use with caution, as it can sometimes resolve installation issues. npm cache cleanforce
npm cache verify	Verifies the integrity of the cache and cleans up corrupted data. npm cache verify
npm config get cache	Gets the cache location. npm config get cache

NPM Configuration

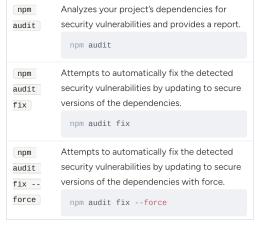
npm config list	Displays the current npm configuration.
	npm config list
npm config get <key></key>	Retrieves the value of a specific configuration key.
	npm config get registry
npm config set <key> <value></value></key>	Sets a configuration key to a specific value.
	<pre>npm config set registry https://registry.npmjs.org/</pre>
npm config delete <key></key>	Deletes a configuration key.
	npm config delete registry

NPM Versioning and Security

Semantic Versioning

NPM uses semantic versioning (semver) to manage package versions. MAJOR.MINOR.PATCH MAJOR: Incompatible API changes MINOR: Add functionality in a backwards compatible manner PATCH: Backwards compatible bug fixes Examples of version ranges: 1.2.3: Exact version ^1.2.3: Compatible with 1.x.x, but not 2.0.0 -1.2.3: Compatible with 1.2.x, but not 1.3.0 Signed than or equal to =: Greater than or equal to *: Any version

Security Audits



Bumping Versions

npm version <new_version></new_version>	Sets the package version to a specific version number. npm version 2.0.0
npm version	Increments the major version number.
npm version	Increments the minor version number. npm version minor
npm version	Increments the patch version number. npm version patch
npm version prerelease	Increments the pre-release version number. npm version prerelease