NPM CHEAT SHEET

WHAT IS NPM?

npm (Node Package Manager) is a popular package manager for JavaScript programming developers. It was initially released in 2010, and is automatically installed by default whenever you install Node.js on your system.

HOW TO INSTALL NPM

npm should be automatically installed when you install Node.js.
Here are some useful commands you can run in your terminal:

1. Check if you have Node.js installed:

```
node -v
```

2. Check if you have npm installed (provided you already have Node.js installed:

```
npm -v
```

3. Update your npm, run the following command:

```
npm install npm@latest -g
```

It's recommended to use a version manager with your Node.js package, such as nodist or NVM.
Read more about downloading and installing npm in the official website.

NPM COMMANDS

`$ npm init`	Create a new npm project/package
`$ npm i <package>`	Install a package
`$ npm u <package>`	Update the package
`$ npm ls`	List all packages
`$ npm install`	Install all the packages mentions in the package.json
`$ npm run <Script>`	Run script in package.json
`$ npm rm <package>`	remove a package
`$ npm audit`	Scan and list all the vulnerabilities in the project
`$ npm audit fix`	Fix found vulnerabilities

10 COMMONLY USED NPM PACKAGES

1. Cloudinary - a cloud service that offers a solution to a web application's entire image management pipeline.
2. Express - one of the best npm packages as it offers a server framework for any type of web application, be it single-page, multi-page or hybrid. For many developers, Express is the standard framework for creating web applications.
3. Nodist - a great npm package for Windows. It is a complete package where you can use different programs like Powershell, CMD, Cygwin, Git bash, and more. However, for beginners, using Nodist can be a big deal.
4. Lodash - a particularly useful package if you're working with lots of numbers, digits, arrays, etc.
5. Browserify - recursively analyze all the require() calls in your app in order to build a bundle you can serve up to the browser in a single <script> tag.
6. Debug - a tiny JavaScript debugging utility modelled after Node.js core's debugging technique. Works in Node.js and web browsers.
7. Grunt - one of the packages where there are several plugins and automation options available to you. With Grunt, your efforts can reduce to a large extent and you can easily shift your focus towards coding.
8. Mocha - a JavaScript test framework, popular among web-developers.
9. Bower - traditionally used for building websites, when only HTML and CSS were essential.
10. Async.js - Many times, JavaScript has become an annoying programming language.
With Async.js, you can get rid of some unnecessary features of JavaScript.One feature that is most difficult about JavaScript is the render-blocking feature.

NPM & JFROG

JFrog provides an end-to-end solution for your npm packages. Easily manage your npm packages using JFrog Artifactory with local, remote and virtual repositories. Scan your npm packages with JFrog Xray, to identify each Javascript file within your npm packages and perform matching and analysis on each one to ensure that your npm application is safe to use.

Did you know? You can also install JFrog CLI with npm in a single command:

```
$ npm install -g jfrog-cli-v2
```