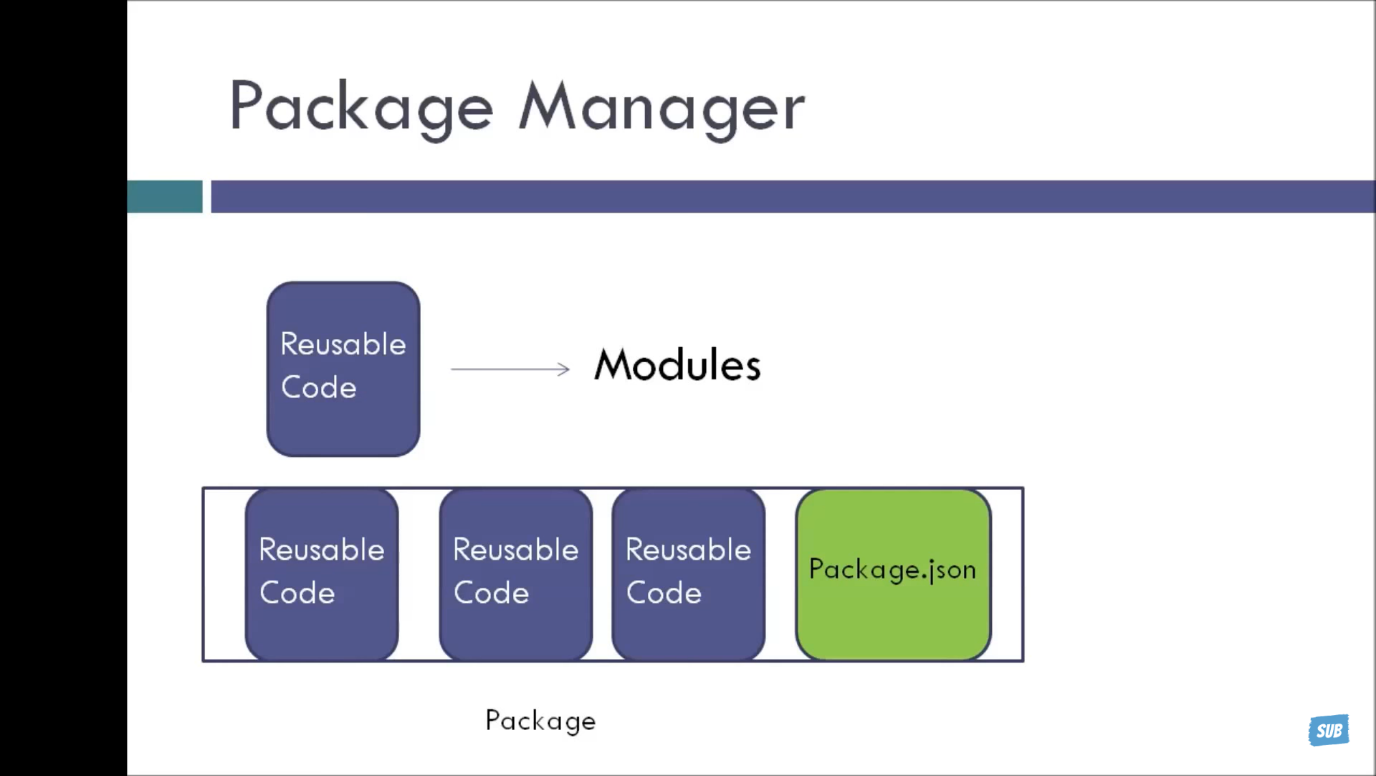
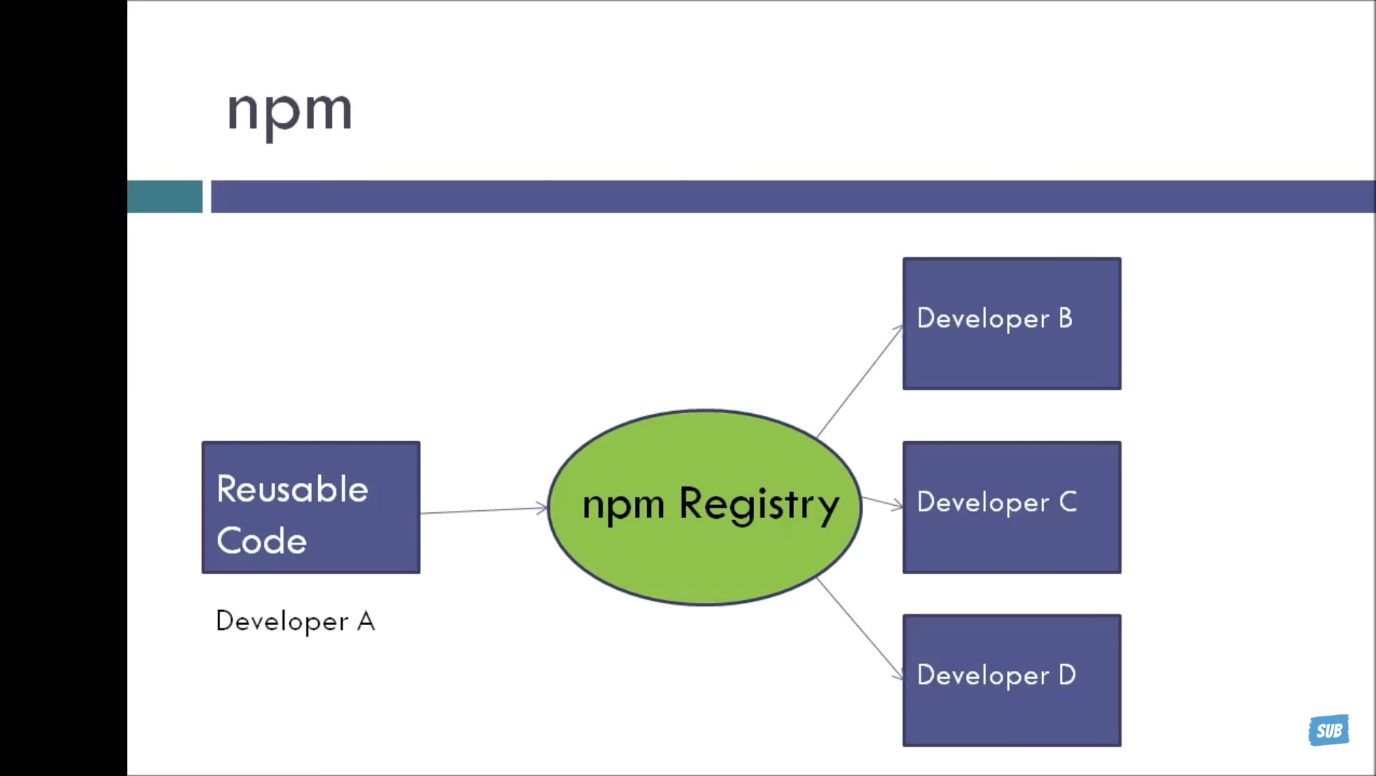
NPM





Creating package.Json:

Text

Description automatically generated

Default package :

Text

Description automatically generated

INSTALLING LOCAL PACKAGE :

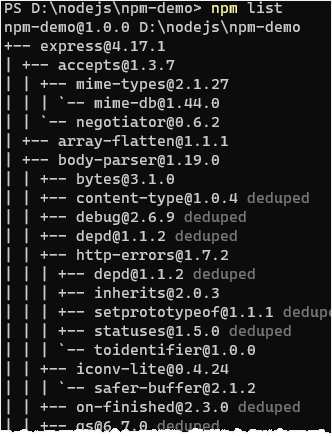
Text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

* To install packages globally we use -g .
* Example : npm install moment -g
* The npm list command outputs installed packages and their dependencies of the current project as a tree



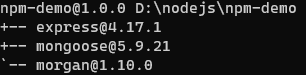
The npm ls is the shorter verison of the npm list command

Listing packages as a tree with a specified depth

To limit the depth of the dependency tree, you use the npm list with the --depth flag.

The following example lists all installed packages without their dependencies:

npm list --depth=0



* Use the npm list to show the installed packages in the current project as a dependency tree.
* Use npm list --depth=n to show the dependency tree with a specified depth.
* Use npm list --prod to show packages in the dependencies.
* Use npm list --dev to show packages in the devDependencies.
* Use npm list --global to list the global packages.
* Use npm list --json to format the installed packages in the JSON format
* Installing from package Json we use :

npm install

* To update a package we use :

npm update Package name

To update all packages just run npm update

To update package globally just run npm update -g

Npm Prune :

This command **removes "extraneous" packages**. If a package name is provided, then only packages matching one of the supplied names are removed. Extraneous packages are those present in the node\_modules folder that are not listed as any package's dependency list.

Npm Scripting :

NPM scripts are **used to automate tasks like minifying CSS, uglifying JavaScript, building project**. NPM scripts are versatile and simple and by learning fewer tools, we can automate tasks in our web project.