# NPM Note Package Manager

* Es un administrador de paquetes de Node.
* Es el registro de paquetes de código más grande.
* Millones de paquetes y dependencias para proyectos
* También es una línea de comandos
* Por ejemplo, react es un paquete de NPM

Hay alternativas a NPM como YARN o PNPM.

**Comandos NPM**

npm init 🡪 Esto creará un package JSON con la descripción del proyecto, las dependencias, los scripts, la licencia, el autor,el título, la versión…

Además de que para omitir hacer el package JSON rellenando se hace con el comando npm init -y o npm init -yes que asigna valores por defecto.

**Dependencia Picocolors**

npm i picocolors,la i es de install

Nos va a permitir estilizar el console.log().

^ 🡪 Este signo se llama karet y lo indica es que cuando haya una versión más actual, se actualizará automáticamente, pero lo recomendable es quitarlo para evitarnos problemas de compatibilidad y uso de recursos.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Debemos tener en cuenta que cuando instalemos dependencias a su misma vez estas pueden tener sus dependencias.

**Desinstalar dependencias**

Mediante el comando npm unistall nombreDependencia

**Tipos de dependencias**

Hay de 2 tipos:

**-Dependencia de producción**

Son aquellas que la aplicación necesita para funcionar

Como Picocolors por ejemplo, que la incluimos de esta manera para que pueda funcionar const pico = require('picocolors');

**-Dependencia de desarrollo**

Para dejar de tener problemas de sintaxis, separaciones, constantes…

Usamos npm install semistandard -D o npm install semistandard –development

Por lo tanto, debemos de indicar que es una dependencia de desarrollo.