

INTRODUCCIÓN A NPM

DEV.F
DESARROLLAMOS(PERSONAS);

dev

NPM

(Node Package Manager)

Administrador de paquetes que se utiliza para instalar y gestionar las dependencias de proyectos Node.js.



Comandos básicos de npm

`npm init` -> iniciar un proyecto npm

`npm install` -> instalar dependencias

`npm uninstall` -> desinstalar un proyecto

DEV.F

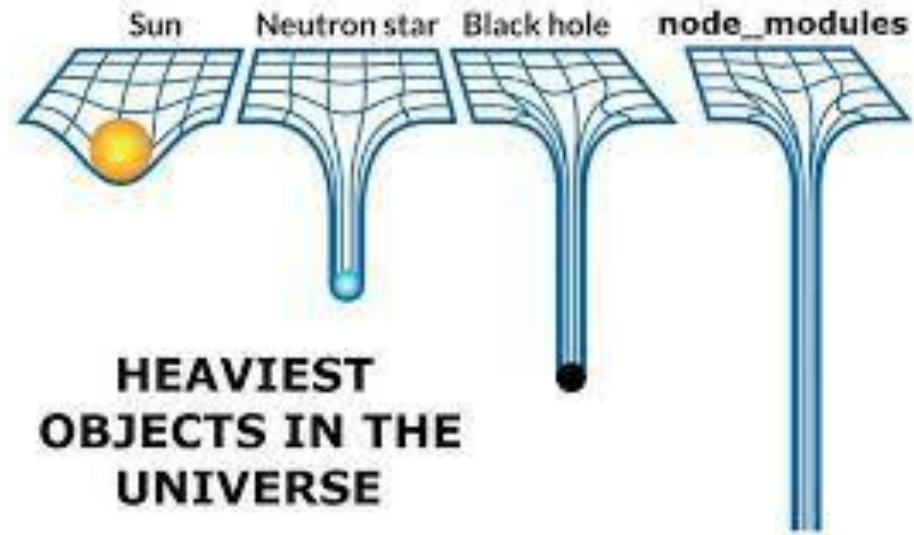
Vamonos al código!



dev

node_modules

Los 'node_modules' son directorios que almacenan las dependencias de un proyecto. Estas dependencias se instalan automáticamente a través de NPM.



package.json vs. package-lock.json

DEV.FX
DESARROLLAMOS(PERSONAS);

dev

package.json

Contiene información sobre el proyecto y sus dependencias.

Dentro de package.json, encontramos información como el nombre del proyecto, versión, autor, scripts personalizados y dependencias.

```
1 {
2   "name": "package-json",
3   "version": "1.0.0",
4   "description": "",
5   "main": "1.fs-node.js",
6   "scripts": {
7     "dev": "nodemon 8.routing.js"
8   },
9   "author": "",
10  "license": "ISC",
11  "dependencies": {
12    "nodemon": "^3.0.0",
13    "picocolors": "1.0.0"
14  },
15  "eslintConfig": {
16    "extends": "standard"
17  }
18 }
```

package-lock.json

Archivo generado automáticamente que registra las versiones exactas de las dependencias del proyecto.

Ayuda a mantener la consistencia de las versiones de las dependencias en diferentes entornos de desarrollo, asegurando que todos los colaboradores tengan las mismas versiones.

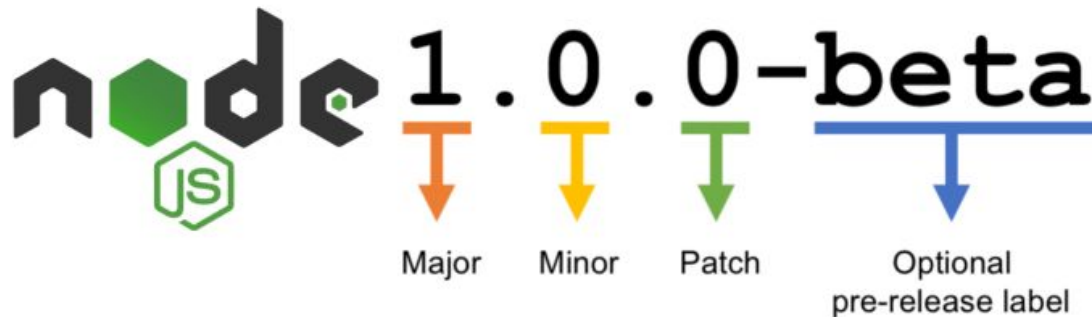
```

1  {
2    "name": "git-prueba",
3    "version": "1.0.0",
4    "lockfileVersion": 3,
5    "requires": true,
6    "packages": {
7      "": {
8        "name": "git-prueba",
9        "version": "1.0.0",
10       "license": "ISC",
11       "dependencies": {
12         "nodeenv": "^3.0.0",
13         "picocolors": "1.0.0"
14       }
15     },
16     "node_modules/abbrev": {
17       "version": "1.1.1",
18       "resolved": "https://registry.npmjs.org/abbrev/-/abbrev-1.1.1.tgz",
19       "integrity": "sha512-nne9/IQI/hzIhY6pdDnb8t7D3JPTKcY00P/zvPSm5pOfK16xuGrGnXn/VCTNFFnfAfZ9/1rtehszU9qctIi0Q="
20     },
21     "node_modules/anymatch": {
22       "version": "3.1.3",
23       "resolved": "https://registry.npmjs.org/anymatch/-/anymatch-3.1.3.tgz",
24       "integrity": "sha512-kR1959vDv18u0970BM7/5H4t23Hg9J6r63du4tJnU6EY4w/2BZD9Cz86nK4jE00O84nqQ1i64HwB00="
25     },
26     "dependencies": {
27       "normalize-path": "^3.0.0",
28       "picomatch": "^2.0.4"
29     },
30     "engines": {
31       "node": ">= 8"
32     }
33   },
34   "node_modules/balanced-match": {
35     "version": "1.0.2",
36     "resolved": "https://registry.npmjs.org/balanced-match/-/balanced-match-1.0.2.tgz",
37     "integrity": "sha512-3o8l1ZrX0jYz7j4Dj2c7j8B888TgU8yRqLgawWeEklMFh2GgRk9lrmESVWzWzL7YJgPQQW69WZ+mh0="
38   },
39   "node_modules/binary-extensions": {
40     "version": "2.2.0",
41     "resolved": "https://registry.npmjs.org/binary-extensions/-/binary-extensions-2.2.0.tgz",
42     "integrity": "sha512-jDlJ6TytD0AqJb3753yBb2Q/0GQ4a05LHdW0uZbD8K1uA/8WUOv70YVgWwWtQ03d9e5uWZUWYU0="
43   },
44   "engines": {
45     "node": ">=8"
46   }
47 },
48 "node_modules/brace-expansion": {
49   "version": "1.1.11",
50   "resolved": "https://registry.npmjs.org/brace-expansion/-/brace-expansion-1.1.11.tgz",
51   "integrity": "sha512-iCuPHDFgrHX7H2vEI/5xpz07zSHB00TugqchY1F/UIEIOglgB7fFozd/hGKKw/CeFFB69GLA=",
52   "dependencies": {
53     "balanced-match": "^1.0.0",
54     "concat-map": "0.0.1"
55   }
56 },
57 "node_modules/braces": {
58   "version": "3.0.2",
59   "resolved": "https://registry.npmjs.org/braces/-/braces-3.0.2.tgz",
60   "integrity": "sha512-b8GMbrXJgbeMXe9b8s7e506sHlV1eWbbON1H0cXklpLjFUrEqx8WwOh7iLh4d9C8XW3THXpZXq3qWZ2=",
61   "dependencies": {
62     "fill-range": "^2.0.1"
63   },
64   "engines": {
65     "node": ">=8"
66   }
67 },
68 "node_modules/chokidar": {
69   "version": "3.5.3",
70   "resolved": "https://registry.npmjs.org/chokidar/-/chokidar-3.5.3.tgz",

```


Semantic Versioning (SemVer)

Sistema de versionado que se utiliza en NPM y otros proyectos. Consiste en tres números: MAJOR.MINOR.PATCH, y sigue reglas específicas para actualizaciones.



El caret (^) en el package.json

El símbolo caret (^) en el archivo package.json permite actualizar una dependencia a su versión más reciente en la misma versión mayor.

Version range

[See examples](#)

List Versions

106 versions found

[View package on npmjs.com](#)

1.0.0	1.1.1	1.8.0	1.14.0	1.17.5
1.0.1	1.2.0	1.8.1	1.14.1	1.18.0
1.0.2	1.2.1	1.9.0	1.14.2	1.18.1
1.0.3	1.3.0	1.9.1	1.14.3	1.18.2
1.0.4	1.3.1	1.9.2	1.14.4	1.18.3

Version range

[See examples](#)

List Versions

83 versions found

[View package on npmjs.com](#)

1.2.0	1.7.0	1.12.3	1.14.8	1.18.2
1.2.1	1.7.1	1.12.4	1.14.9	1.18.3
1.3.0	1.7.2	1.12.5	1.14.10	1.18.4
1.3.1	1.7.3	1.12.6	1.14.11	1.18.5
1.3.2	1.8.0	1.12.7	1.14.12	1.18.6

<https://semver.npmjs.com>