

Yarn

DEV.F
DESARROLLAMOS(PERSONAS);

dev

Yarn

Es un gestor de dependencias de javascript (alternativa a npm), con mejoras de velocidad y seguridad en comparación con el cliente npm.

- Creado en 2016 por ingenieros de facebook y google.
- Es más amigable en su uso (sintaxis y colores).
- Utiliza npm y depende de node.
- Npm sigue siendo el repositorio central de paquetes, solo se cambia el cliente (gestor de paquetes/dependencias).



Ventajas de Yarn

- **Velocidad:** La mayoría de la instalación de paquetes por npm toma minutos, en yarn toma segundos.
- **Seguridad:** Yarn verifica la integridad de cada paquete y comprueba que sea seguro.
- **Fiabilidad:** La red de npm suele tener issues, yarn tiene un CDN de mayor fiabilidad.
- **Múltiple registro:** Tiene acceso al registry de bower y npm, mientras que npm solo al de npm.

Docs

Para más sobre Yarn checar la siguiente [docs](#).

Comandos Yarn

Comando	Descripción
<code>npm install -g yarn</code>	Instalar globalmente yarn.
<code>yarn --version</code>	Validar versión de yarn.
<code>yarn init</code>	Inicializar un proyecto yarn.
<code>yarn add <package-name></code>	Agregar una dependencia.
<code>yarn install</code>	Instalar una dependencias.
<code>yarn publish</code>	Publicar un paquete.

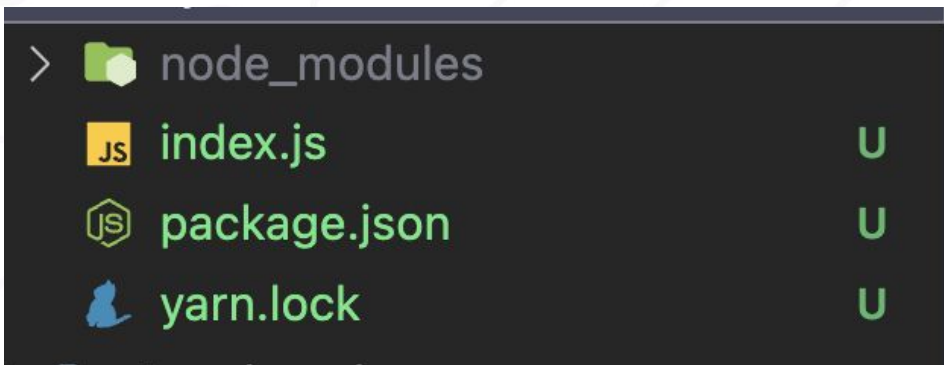
Scaffold Yarn

DEV.F
DESARROLLAMOS(PERSONAS);

dev

Estructura de proyecto yarn

- **node_modules:** Carpeta donde se instalan las dependencias de un proyecto npm, normalmente esta carpeta se agrega al .gitignore.
- **package.json:** Guardan las dependencias y los comandos de node.
- **yarn.lock:** Guarda un snapshot de las dependencias que se instalaron en un determinado momento.



Pasos para crear un proyecto yarn

1. En la carpeta raíz del proyecto ejecutar: **yarn init**.
2. Instalar las dependencias necesarias con: **yarn add <package-name>**
3. Agregar los scripts necesarios al package.json
4. Ejecutar el proyecto.

Ejemplo [ejercicio 2-yarn](#)

```
yarn init  
yarn add express  
yarn add colors  
yarn run start
```