

## Comandos

### Instalar desde package.json:

`pnpm install`

### Que el archivo de bloqueo no se actualice:

`pnpm install --frozen-lockfile`

### Añadir paquete:

`pnpm add <paquete> [--global]`

### Añadir paquete como dependencies:

`pnpm add <paquete> --save-prod`

### Añadir paquete como peerDependencies:

`pnpm add <paquete> --save-peer`

### Añadir paquete como devDependencies:

`pnpm add <paquete> --save-dev`

### Añadir paquete como optionalDependencies:

`pnpm add <paquete> --save-optional`

### Añadir versión exacta:

`pnpm add <paquete> --save-exact`

### Desinstalar paquete:

`pnpm uninstall <paquete> [--global]`

### Actualizar paquete:

`pnpm update [<paquete>] [--latest] [--global]`

### Actualizar interactiva:

`pnpm update --interactive`

### Vincular paquete local:

`pnpm link <directorio>`

### Listar todos los paquetes del nivel superior:

`pnpm list --depth 0`

### Auditar las dependencias vulnerables:

`pnpm audit [fix]`

### Lista de paquetes obsoletos:

`pnpm outdated`

### Inicializar:

`pnpm init`

### Inicio/Cierre de sesión:

`pnpm <login | logout>`

### Ejecutar scripts:

`pnpm [run] <script>`

### Ejecutar test:

`pnpm test`

### Crear paquete build:

`pnpm build`

### Publicar:

`pnpm publish`

### Despublicar:

`pnpm unpublish <paquete>[@#.#.#]`

### Hacer un paquete obsoleto:

`pnpm deprecate <paquete>[@#.#.#] <mensaje>`

### Lista de configuración:

`pnpm config list`

### Configurar --save-default como predeterminado:

`pnpm config set save-exact true`

### Configurar ~ por defecto en lugar de ^:

`pnpm config set save-prefix '~'`

### Ejecutar paquete:

`pnpm dlx <paquete>`