

## Comandos Artisan

Los comandos Artisan son la interfaz de línea de comandos de Laravel para automatizar tareas, crear código (modelos, controladores, migraciones), gestionar la base de datos (migrar, sembrar), servir la aplicación y realizar tareas de mantenimiento

### Comandos esenciales

- `php artisan list`: Muestra todos los comandos disponibles.
- `php artisan serve`: Inicia el servidor de desarrollo local de Laravel.
- `php artisan tinker`: Abre una consola interactiva (REPL) para ejecutar código PHP en el contexto de tu aplicación.
- `php artisan route:list`: Muestra todas las rutas definidas en la aplicación.

### Generación de código (`make:`)

- `php artisan make:model <Nombre>`: Crea un nuevo modelo.
- `php artisan make:controller <Nombre>`: Crea un nuevo controlador.
- `php artisan make:migration <nombre_migracion>`: Crea un archivo de migración para la base de datos.
- `php artisan make:seeder <Nombre>`: Crea un archivo para poblar la base de datos con datos.
- `php artisan make:component <Nombre>`: Crea un componente Blade.

### Gestión de base de datos

- `php artisan migrate`: Ejecuta las migraciones pendientes.
- `php artisan migrate:fresh`: Elimina todas las tablas y luego ejecuta las migraciones.
- `php artisan migrate:refresh`: Deshace todas las migraciones y luego las vuelve a ejecutar.
- `php artisan db:seed`: Ejecuta los *seeders* para poblar la base de datos.

## Mantenimiento y caché

- `php artisan down`: Pone la aplicación en modo de mantenimiento.
- `php artisan up`: Saca la aplicación del modo de mantenimiento.
- `php artisan config:cache`: Cachea los archivos de configuración.
- `php artisan cache:clear`: Limpia la caché de la aplicación.

## Ejemplos de uso combinado

- `php artisan make:model User -a`: Crea un modelo `User` junto con su controlador, migración y factoría.
- `php artisan migrate:fresh --seed`: Limpia la base de datos y la rellena con datos de prueba.