

## Git en Eclipse

Para poder trabajar con Git en Eclipse es necesario iniciar el repositorio desde las opciones integradas en el menú de herramientas del IDE, una vez creada el repositorio se podrá realizar comandos de Git (add, commit, push, pull, fetch...).

Para publicar o Clonar repositorio desde una Plataforma de repositorio como GitHub, es necesario crear un Token (Ficha o clave con permiso de acceso) en <https://github.com/settings/tokens> junto con el usuario o correo de GitHub se podrá realizar una conexión a GitHub desde Eclipse.

## Git en NetBeans

Para poder trabajar con Git en NetBeans es necesario iniciar el repositorio desde las opciones integradas en el menú de herramientas del IDE, una vez creada el repositorio se podrá realizar comandos de Git (add, commit, push, pull, fetch...).

Para publicar o Clonar repositorio desde una Plataforma de repositorio como GitHub, es necesario crear un Token (Ficha o clave con permiso de acceso) en <https://github.com/settings/tokens> junto con el usuario o correo de GitHub se podrá realizar una conexión a GitHub desde NetBeans.

## Git en IntelliJ IDEA

Trabajar Git en este IDE es más cómodo que las dos antes mencionadas ya que este te permite iniciar sesión con la cuenta de GitHub, la cual se podrá clonar, sincronizar (pull & push) de manera más sencilla y en su mayoría no genera error de conexión de IDE-GitHub

## Git en VS Code

Considero que es la forma más optimizada de trabajar GitHub-Editor Texto, permite iniciar sesión de GitHub desde el mismo VS Code y realizar comandos de git de forma amigable tales como add, pull, push, commit, fetch, clone... con tan solo unos cuantos clicks.