

Lista de comandos de Git

- **git config --global user.name "Nombre de usuario".** Configurar nombre de usuario.
- **git config --global user.email (email).** Configurar email.
 - Usando estos comandos sin usar el global, cambiaríamos sólo el usuario y email del proyecto actual.
- **git config --list.** Comprobar tu configuración.
 - **--system** (todos los usuarios), **--global** (usuario específico).
 - Valor específico **git config --global user.name**.
- **git init.** Respaldo de un directorio.
- **git status (-s o --short, para info resumida)** Ver estado del working directory.
- **git add (nombre fichero).** Hacer un seguimiento (pasar a staging área).
 - Si se le pone **git add .** los añade todos.
- **git commit -m "Texto de la subida".** Subir fichero al repositorio.
 - **git commit -a -m texto_subida.** Sube directamente sin pasar por la staging área.
 - **git commit --amend -m texto_subida.** Modifica la última subida.
- **git log --oneline.** Ver las revisiones del fichero en el repositorio.
 - **git log nombre_fichero.** Versiones en las que ha participado.
 - **git log --stat.** Todas las versiones con mucha información.
 - **git log --name-status.** Ficheros modificados en cada versión
- **nano .gitignore.** Crear un fichero en el que se pueden introducir las extensiones que queremos que git ignore.
- **git rm --cached <archivo>.** Quitar un archivo de la staging área.
- **git clone url_repositorio carpetaMiMáquina.** Descargar el repositorio de la url en la carpeta indicada (carpetaMiMáquina).
- **git diff.** Muestra la diferencia con lo que hay en el repositorio.
 - **git diff staged.** Los que se ha preparado y serán incluidos en la próxima confirmación.
- **Git restore [--worktree] <file>.** Sobrescribe únicamente la copia del trabajo, pero no la del index (Se puede omitir --worktree).
- **git restore --staged <file>.** Sobrescribe nuestro archivo de índice con el presente HEAD del repositorio local.
- **git restore --staged --worktree --source HEAD <file>.** Sobrescribe tanto el archivo índice como la copia del trabajo.
- **git remote add origin url_gitHub.** Conectar nuestro repositorio local a GitHub.
- **git branch nombre_rama.** Crear una rama nueva.
 - **git branch.** Visualizar las ramas creadas.
- **git checkout nombre_rama.** Cambiar a la rama nombre_rama.
- **git branch -d nombre_rama.** Elimina la rama nombre_rama.
- **git push -u origin main.** Subir la rama main a GitHub.
- **git pull origin main.** Trae de GitHub a nuestro repositorio los ficheros.
- **git tag nombre_etiqueta.** Crea una nueva etiqueta.
- **git push --tags.** Subir las etiquetas a GitHub.
- **git clone url_github.** Clona el proyecto de GitHub al repositorio local.
- **git merge nombre_rama_fusionar.** Fusionar una rama. Nos tenemos que trasladar a la rama a la que fusionamos antes.

- **git merge main.** Actualizar una rama con la master.