

Configuracion de Repositorios

Iniciar un repositorio local.

```
>_ git init
```

Definir el nombre de usuario que trabajará en el repositorio local.

```
>_ git config user.name [nombre de usuario que aparece en GitHub]
```

Verificar el nombre de usuario.

```
>_ git config user.name
```

Definir el correo electrónico de quien trabajará en el repositorio local.

```
>_ git config user.email [correo electrónico registrado en GitHub]
```

Verificar el nombre de usuario.

```
>_ git config user.email
```

Establecer usuario global para todos los repositorios futuros

```
>_ git config --global user.name [nombre de usuario que aparece en GitHub]
```

Confirmar correo de usuario global para todos los repositorios futuros

```
>_ git config --global user.email [correo electrónico registrado en GitHub]
```

Verificar la conexión

```
>_ git remote add origin [URL del repositorio remoto]
```

Verificar el nombre de usuario.

```
>_ git remote -v
```

Eliminar conexión a repositorio remoto.

```
>_ git remote remove origin
```

Eliminar el trackeo del repositorio local actual.

```
>_ rm -rf .git
```

Modificar el nombre de la rama principal por defecto a main.

```
>_ git config --global init.defaultBranch main
```

Manejo de archivos versionados

Agregar todos los archivos del directorio actual

```
>_ git add .
```

Agregar un archivo específico del directorio actual

```
>_ git add . [nombre del archivo]
```

Conocer el estado de los archivos del repositorio.

```
>_ git status
```

Pasa los docs de Warp Directory a Staging Srea, dejando un mensaje o comentario "x"

```
>_ git commit -m "mensaje"
```

Sacar todos los archivos del Warp Directory

```
>_ git reset .
```

Conocer el historial de todos los commit realizados, con fecha, nombre y mensaje.

```
>_ git log
```

Ver el historial más resumido

```
>_ git log --oneline
```

Ver el historial en forma gráfica.

```
>_ git log --graph
```

Modificar el nombre de una rama.

```
>_ git branch -M [nuevo nombre]
```

Regresar en el tiempo a un commit antiguo.

```
>_ git checkout [SHA del commit]
```

Modificar el mensaje del último commit.

```
>_ git commit --amend
```

Carga y descarga archivos

Clona el repositorio remoto en un directorio local.

```
>_ git clone [URL del repositorio remoto]
```

Envía los archivos del Stading Area al Repository.

```
>_ git push origin main/master
```

Descarga las últimas versiones del repositorio remoto en repsitorio local.

```
>_ git pull
```

Ramas o Branches

Enumera todas las ramas del repositorio.

```
>_ git branch
```

Crea una nueva rama con el nombre digitado.

```
>_ git branch [nombre de la rama]
```

Elimina una rama específica.

```
>_ git branch -d [nombre rama]
```

Elimina de manera forzada una rama específica.

```
>_ git branch -D [nombre rama]
```

Se mueve a la rama especificada.

```
>_ git checkout [nombre rama]
```

Modificar el nombre de una rama.

```
>_ git branch -M [nuevo nombre]
```

Subir archivos commiteados a la rama mencionada.

```
>_ git push origin [nombre rama]
```

Descargar las actualizaciones de la rama hechas en el el Repository.

```
>_ git pull origin [nombre rama]
```

Mezclar el contenido de la rama actual con e la rama mencionada.

```
>_ git merge [nombre rama]
```