

Pasos generales para un proyecto

Abrir en la terminal la carpeta que contiene el proyecto.

```
$ cd folder_with_project
```

Crear un repositorio local en la carpeta del proyecto.*

```
$ git init
```

Añadir al repositorio local los archivos que se encuentran en la carpeta del proyecto.

```
$ git add .  
$ git add file_name.ext
```

Realizar el commit de los archivos que están en staging area.

```
$ git commit -m "Commit message"
```

Si se quiere subir a un repositorio remoto (Github):

Conectar el repositorio local con un repositorio remoto (en GitHub)**

```
$ git remote add origin repository_url.git
```

Subir los últimos cambios del repositorio local al repositorio remoto

```
$ git push origin branch_name
```

Repositorios

Crear un repositorio*

```
$ git init
```

Ver la situación actual de los archivos

```
$ git status
```

Añadir un archivo al repositorio

```
$ git add file_name.ext
```

Añadir todos los archivos que contiene la carpeta

```
$ git add .  
$ git add -A
```

Mostrar las diferencias de archivos que no se han enviado aún al staging area

```
$ git diff
```

Borrar el caché de un archivo**

```
$ git rm --cached file_name.ext
```

Repositorios remotos

Conectar un repositorio local con un repositorio remoto (en GitHub) *

```
$ git remote add origin repository_url.git
```

Listar los repositorios remotos a los que se está conectado

```
$ git remote -v
```

Cambiar la url del repositorio remoto al que se está conectado

```
$ git remote set-url origin new_repository_url
```

Actualizar el repositorio local para que tenga los últimos cambios del repositorio remoto

```
$ git pull origin master
```

```
$ git pull origin branch_name
```

Subir los últimos cambios del repositorio local al repositorio remoto

```
$ git push -u origin master
```

```
$ git push origin branch_name
```

Crear una copia al repositorio local, clonando un repositorio remoto

```
$ git clone repository_url
```

Ramas

Crear una nueva rama (branch)

```
$ git branch new_branch_name
```

Ver todas las ramas del repositorio

```
$ git branch
```

Cambiar de rama a donde se quiere subir el archivo

```
$ git checkout branch_name
```

Crear una nueva rama y "cambiarse" a ella

```
$ git checkout -b new_branch_name
```

Volver a la rama master

```
$ git checkout master
```

Unir una rama con la rama activa actualmente

```
$ git merge merging_branch_name
```

Borrar una rama

```
$ git branch -d deleting_branch_name
```

Forzar el borrado de una rama (force-delete)

```
$ git branch -D deleting_branch_name
```

Configuraciones

Configurar el nombre de usuario para todo el sistema

```
$ git config --global user.name "my_username"
```

Configurar el nombre de usuario solo para el repositorio actual

```
$ git config user.name "my_username"
```

Configurar el email para todo el sistema

```
$ git config --global user.email "my_email"
```

Configurar el email solo para el repositorio actual

```
$ git config user.email "my_email"
```

Habilitar la colorización en la línea de comando

```
$ git config --global color.ui auto
```

Eliminar todas las configuraciones globales realizadas

```
$ git config --global --unset-all
```

Commit - Files

Realizar el commit de los archivos que están en staging area

```
$ git commit -m "Commit message"
```

Mostrar todos los commits realizados

```
$ git log
```

```
$ git log --oneline
```

Revertir el commit a uno anterior*

```
$ git revert commit_ID
```

Deshacer todos los commits que se hicieron después del commit_ID, preservando los cambios localmente*

```
$ git reset commit_ID
```

Enumerar el historial de versión para un archivo, incluidos los cambios de nombre

```
$ git log --follow file_name
```

Cambiar el nombre de un archivo

```
$ git mv original_file_name new_file_name
```