



Certified Tech Developer
The Ultimate Degree

Comandos de Git

Configuración del repositorio local	
\$ git init	Crea el repositorio local en el directorio actual.
\$git config user.name "mi_usuario". Opcional: -global	Configura el nombre de usuario del administrador del repositorio local.
\$git config user.email "correo@mail.com". Opcional: -global	Sincroniza el correo electrónico de GitHub con el repositorio local.

Trabajo en el repositorio local	
\$ git add <archivo>	Añade un archivo al seguimiento (<i>tracked</i>) de Git. Si sufre modificaciones, saldrá de esta área.
\$ git add .	Añade todos los archivos del directorio actual al seguimiento.

\$ git commit -m "mensaje"	Confirma todos los archivos que estaban en seguimiento, creando un <i>commit</i> . El <i>commit</i> representa el guardado de una versión importante del trabajo en el tiempo.
\$git status	Indica el estado de los archivos presentes en el directorio del repositorio local.
\$ git log	Muestra el historial de commits ordenado cronológicamente, con detalles específicos de cada uno.



Trabajo con el repositorio remoto

\$ git remote add origin <link.git>	Linkea el repositorio local, con el remoto de GitHub.
\$ git remote -v	Responderá con dos link: fetch y push. Permite comprobar el éxito del linkeo.
\$ git clone <link.git>	Linkea el repositorio remoto con el local, descargando en la PC los archivos existentes en la nube.
\$ git push origin main	Sube los archivos al repositorio remoto, a la rama principal (<i>main</i>).
\$ git pull origin main	Actualiza los archivos del repositorio local con respecto al remoto, en la rama principal.
\$ git fetch	Verifica si hay actualizaciones existentes en el repositorio remoto.
\$ git reset -MODE <commit>	Restablece el estado actual del directorio a un estado específico. Existen varios atributos

	para ello, usando --MODE.
\$ git revert	Revierte los cambios en un commit.

Trabajo con ramas (branches) de la main.

\$ git branch	Enumera todas las ramas del repositorio.
\$ git branch <branch>	Crea una nueva rama.
\$ git branch -d <branch>	Elimina la rama. Git lo evitará si hay cambios no fusionados con la main.
\$ git branch -D <branch>	Fuerza la eliminación de la rama.
\$ git checkout <branch>	Permite moverse de una rama a otra.
\$ git push/pull origin <branch>	Sube o actualiza los archivos trabajados en la rama.
\$ git merge	Fusiona los cambios realizados en la rama con la HEAD (usualmente la <i>main</i>), que será la rama de recepción.