**Cómo funcionan los commits**

**Tabla de contenido**

* ¿Qué es un commit?
* Creación de un commit
* Visualización de commits
* Commits en GitHub
* Conclusión
* Recursos

Un aspecto fundamental de Git es el concepto de “commit”, que es esencialmente una instantánea de tu código en un punto específico del tiempo.

**¿Qué es un commit?**

Un commit en Git es un cambio que se ha guardado o “comprometido” en tu repositorio. Cada commit tiene un identificador único (llamado hash SHA), y contiene información sobre los cambios realizados, quién los hizo, y cuándo se hicieron.

**Creación de un commit**

Para crear un commit, primero debes hacer cambios en tu repositorio. Estos cambios pueden ser cualquier cosa, desde agregar un nuevo archivo hasta modificar una línea de código. Una vez que hayas hecho tus cambios, puedes usar el comando git add para agregar estos cambios al “área de preparación” (también conocida como “índice”).

git add .

Una vez que tus cambios están en el área de preparación, puedes usar el comando git commit para crear un nuevo commit. Este comando abrirá un editor de texto donde puedes escribir un mensaje de commit, que es una descripción breve de los cambios que has hecho.

git commit -m "Tu mensaje de commit"

**Visualización de commits**

Puedes ver la historia de commits de tu repositorio utilizando el comando git log. Este comando mostrará una lista de todos los commits en tu repositorio, empezando por el más reciente. Cada entrada en el log incluye el hash SHA del commit, el autor del commit, la fecha del commit y el mensaje del commit.

git log

**Commits en GitHub**

Cuando trabajas con GitHub, los commits que haces localmente pueden ser “empujados” a tu repositorio de GitHub con el comando git push. Esto permite que otros vean y colaboren en tus cambios. Del mismo modo, puedes “jalar” los cambios de otros a tu repositorio local con el comando git pull.

git push origin main

git pull origin main

**Conclusión**

Los commits son una parte esencial del flujo de trabajo en Git y GitHub. Permiten un seguimiento detallado de los cambios, facilitan la colaboración y ayudan a prevenir la pérdida de trabajo. Al entender cómo funcionan los commits, puedes utilizar GitHub de manera más efectiva para gestionar y colaborar en proyectos de desarrollo de software.

**Recursos:**

Qué es y cómo realizar un Commit: <https://www.youtube.com/watch?v=j9zAL52wuLg>

Creando tu primer commit:

<https://www.youtube.com/watch?v=UJGKWMHX038>

Cómo revertir un Commit (revert, reset):

<https://www.youtube.com/watch?v=HZ1c25OIX4o>